

**PERANCANGAN PROGRAM APLIKASI PENJADWALAN
PROSES PRODUKSI BUKU TULIS PADA JALUR MESIN 321
DI PT. SOLO MURNI DENGAN MENGGUNAKAN
PENDEKATAN *DRUM-BUFFER-ROPE (DBR)***

Skripsi



**AJENG RACHMI PUWARA SITHORESMI
I 1305019**

**JURUSAN TEKNIK INDUSTRI FAKULTAS TEKNIK
UNIVERSITAS SEBELAS MARET
SURAKARTA
2010**

BAB I PENDAHULUAN

Pada bab ini akan dijelaskan hal-hal yang menjadi latar belakang dan permasalahan dari penelitian yang dilakukan di PT. SOLO MURNI, tujuan yang ingin dicapai, manfaat yang dapat diberikan pada perusahaan, serta batasan dan asumsi yang digunakan selama penelitian tugas akhir.

1.1. Latar Belakang Penelitian

Bottleneck dalam sistem penjadwalan produksi merupakan suatu sumber daya terbatas yang terdapat pada sistem sehingga dapat mempengaruhi penjadwalan produksi (Sipper dan Bulfin, 1997). Permasalahan *bottleneck* ini sering terjadi di perusahaan. Hal ini bisa terjadi karena adanya mesin yang memiliki kapasitas terbatas, pekerja atau karyawan yang terbatas jumlahnya, adanya kebijakan (*policy*) untuk tidak diperbolehkan lembur ataupun karena permintaan pasar yang kurang dari kapasitas produksi (Sipper dan Bulfin, 1997). *Bottleneck* harus segera diatasi agar perusahaan memiliki daya kompetisi yang baik di era persaingan bisnis saat ini yang semakin menggilu.

Hal diatas juga terjadi di PT. Solo Murni sebagai perusahaan percetakan yang menghasilkan produk *stationary* dengan kualitas "Take The Best" atau "Pilihan Terbaik". *Bottleneck* yang terjadi di sana karena adanya stasiun kerja Vurnish Cover yang memiliki kapasitas paling rendah dari stasiun kerja yang lainnya. Akibatnya, di stasiun kerja tersebut terjadi penumpukan *work in process* (WIP) kertas. Penumpukan WIP kertas di lantai produksi ini, akan menimbulkan masalah kerusakan pada kertas karena ditumpuk terlalu lama, sehingga perlu dilakukan pengaturan agar masalah ini dapat teratasi. Salah satu cara pengaturan tersebut dengan melakukan penjadwalan produksi yang tepat dan akurat dengan memperhatikan aliran proses dan kapasitas yang tersedia. Jadwal produksi tersebut dibuat oleh bagian Departemen PPIC PT. Solo Murni.

Produk di PT. Solo Murni dikelompokkan menjadi enam golongan yaitu *school supplies*, *office supplies*, *writing papers*, *envelopes*, *gift wrapping set*, dan *others*. Produk andalannya adalah produk buku tulis yang termasuk golongan *school supplies*. Produk buku tulis ini merupakan produk yang *repetitive*, yaitu

tidak melibatkan unsur perancangan dalam proses manufaktur dan pemesanan model yang sama dimungkinkan terjadi dalam waktu singkat (PT. Solo Murni, 2009).

Pada proses produksi buku tulis (*school supplies*) tersebut dibagi menjadi dua jalur produksi yaitu jalur mesin 321 (melalui mesin jahit kawat miller martini) dan jalur tebalan (melalui mesin jahit benang dan mesin monoblok) (PT. Solo Murni, 2009). Diantara kedua jalur tersebut, pada jalur mesin 321 itu yang terjadi penumpukan *work in process* (WIP) kertas. Oleh karena itu maka pada jalur mesin 321 ini dilakukan pengaturan agar masalah yang terjadi di jalur tersebut dapat diatasi.

Selama ini metode penjadwalan yang digunakan adalah *forward scheduling* (PT. Solo Murni, 2009). Metode *forward scheduling* yaitu penjadwalan produksi dilakukan mulai dari stasiun kerja awal berurut-urut hingga stasiun kerja akhir. Penjadwalan di perusahaan dilakukan dengan menjadwalkan order-order yang masuk yang diperhitungkan per bagian kerja sesuai target perminggu. Bagian-bagian tersebut adalah bagian D303 untuk proses cetak isi, cetak cover, vurnish cover dan embosst cover; bagian D304 untuk proses lipat isi dan potong cover; dan bagian D305 untuk proses mesin 321. Karena penjadwalan dihitung perbagian kerja sesuai target per minggu, maka menyebabkan hasil penjadwalan tidak rinci padahal dalam bagian kerja tersebut terdapat beberapa proses yang dilakukan. Selain itu juga menimbulkan *flowtime* penyelesaian order yang besar. Untuk mengatur hal tersebut, maka perlu metode yang dapat meminimasi *mean flowtime* order pada penjadwalan produksi buku tulisnya. Seperti pernyataan Baker (1974) bahwa meminimasi *mean flowtime* berarti juga akan meminimasi WIP. Selain mengendalikan WIP, untuk mengatur penjadwalan produksi buku tulis dengan sistem *bottleneck* juga diperlukan suatu metode penjadwalan produksi yang mampu mengatur hal tersebut di rantai produksi.

Penjadwalan produksi dengan sistem *bottleneck* untuk produk buku tulis yang melewati jalur mesin 321 dapat dipecahkan dengan pendekatan *Drum-Buffer-Rope* (DBR) (Sipper dan Bulfin, 1997). Penjadwalan produksi buku tulis dengan pendekatan DBR ini digunakan karena pendekatan tersebut mengakomodasi keberadaan SKB dan menjadikannya sebagai titik kontrol dalam

penjadwalan produksi. Dalam pendekatan *DBR* ini, penjadwalan produksi buku tulis yang pertama dilakukan di SKB dahulu, selanjutnya untuk stasiun kerja sebelum SKB (*SK non-bottleneck upstream*) dan stasiun kerja sesudah SKB (*SK non-bottleneck downstream*) dijadwalkan sesuai dengan jadwal di SKB. Penjadwalan produksi pada SKNB *upstream* menggunakan *backward scheduling* dan SKNB *downstream* menggunakan *forward scheduling*.

Untuk menerapkan penjadwalan produksi buku tulis maka dibutuhkan suatu program aplikasi penjadwalan produksi yang dapat membantu Departemen PPIC di PT. Solo Murni. Program aplikasi bertujuan untuk mempermudah dan mempercepat proses penjadwalan produksi buku tulis, karena penjadwalan produksi dengan *DBR* ini sulit untuk diterapkan secara manual. Program aplikasi tersebut dibuat menggunakan rancangan model penjadwalan produksi buku tulis di jalur mesin 321 dengan menggunakan pendekatan *DBR*.

1.2. Perumusan Masalah

Berdasarkan latar belakang yang membahas bahwa proses produksi buku tulis pada jalur mesin 321 terdapat *bottleneck* di stasiun kerja Vurnish, selanjutnya penelitian ini akan difokuskan pada permasalahan bagaimana merancang suatu program aplikasi yang dapat membantu dalam penjadwalan produksi buku tulis jalur mesin 321 PT. Solo Murni dengan menggunakan pendekatan *Drum-Buffer-Rope*.

1.3. Tujuan Penelitian

Adapun tujuan yang ingin dicapai dari penelitian ini adalah sebagai berikut:

1. Merancang algoritma penjadwalan produksi buku tulis yang melewati jalur mesin 321 dengan menggunakan pendekatan *DBR* untuk meminimasi *bottleneck* yang terjadi jalur tersebut.
2. Membuat program aplikasi penjadwalan produksi yang dapat membantu proses penjadwalan produksi buku tulis jalur mesin 321 PT. Solo Murni dengan menggunakan pendekatan *DBR*.

1.4. Manfaat Penelitian

Penelitian ini diharapkan mampu memberikan manfaat kepada PT. Solo Murni. Manfaat yang akan diperoleh dari hasil penelitian ini adalah:

1. Menghasilkan jadwal produksi buku tulis yang lebih baik, yang mengakomodasi keberadaan SKB.
2. Mempermudah dalam penjadwalan produksi buku tulis karena ada program aplikasi.
3. Menghasilkan jadwal produksi buku tulis yang memiliki *mean flowtime* yang rendah.

1.5. Batasan Penelitian

Batasan yang digunakan dalam penelitian ini adalah:

1. Data order yang digunakan dalam penelitian tugas akhir ini merupakan data pada bulan April sampai dengan Mei tahun 2009 di PT. Solo Murni.
2. Implementasi program aplikasi penjadwalan produksi buku tulis yang dirancang tidak dibahas.

1.6. Asumsi-asumsi

Asumsi-asumsi yang digunakan dalam penelitian ini adalah:

1. Waktu yang diperlukan untuk berubah (*setup*) dari satu model ke bentuk model yang lainnya sangat kecil, dan model yang berbeda diproduksi pada lini produksi yang sama, sehingga waktu *setup*-nya termasuk di dalam waktu proses.
2. Kenyataan di lantai produksi menunjukkan bahwa terdapat jam kerja lembur (*overtime*), yaitu pada hari Minggu. Penggunaan jam kerja lembur ini jarang dilakukan dan dilakukan sesuai dengan kebutuhan. Pemanfaatan jam lembur mengakibatkan penambahan biaya lembur. Penggunaan jam kerja lembur diusahakan seminimal mungkin, sebaiknya menggunakan alternatif pemanfaatan jam kerja longgar serta pengalihan order ke mesin yang menganggur.
3. Kebijakan yang dilakukan oleh PT. Solo Murni terhadap keberadaan order sisipan yaitu jika ada order sisipan ketika proses produksi berlangsung, maka dijadwalkan pada minggu berikutnya. Kebijakan tersebut bertujuan

agar tidak mengganggu proses produksi yang telah dijadwalkan di lantai produksi.

1.7. Sistematika Penulisan

Penulisan laporan penelitian tugas akhir ini disusun secara sistematis menjadi enam bab, seperti diuraikan dibawah ini.

BAB I PENDAHULUAN

Bab ini membahas tentang latar belakang yang diangkat dalam penelitian, perumusan masalah, tujuan penelitian, manfaat penelitian, pembatasan masalah, penetapan asumsi-asumsi serta sistematika yang digunakan dalam penelitian.

BAB II TINJAUAN PUSTAKA

Bab ini membahas tentang perusahaan maupun teori-teori yang digunakan sebagai landasan penelitian baik dari buku teks, jurnal, maupun berbagai sumber pustaka lainnya.

BAB III METODOLOGI PENELITIAN

Pada bab ini dijelaskan mengenai langkah-langkah pemecahan masalah pada penelitian yang dilakukan.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Pada bab ini akan dilakukan pengumpulan dan pengolahan data yaitu meliputi penyajian data yang diperoleh, menganalisa data tersebut yang langsung dipakai untuk memecahkan persoalan.

BAB V ANALISIS DAN INTERPRETASI HASIL

Bab ini membahas tentang analisis dan interpretasi hasil dari pengolahan data pada penelitian.

BAB VI KESIMPULAN DAN SARAN

Bab ini merupakan bagian akhir dari keseluruhan isi utama tugas akhir. Bab ini akan membahas simpulan hasil yang diperoleh, serta usulan atau saran yang memberikan manfaat diajukan kepada pihak yang berkepentingan.

BAB II TINJAUAN PUSTAKA

Bab ini membahas teori-teori yang digunakan untuk merancang suatu alat bantu program aplikasi penjadwalan produksi buku tulis jalur mesin 321 di departemen produksi PT. Solo Murni dengan menggunakan pendekatan *Drum-Buffer-Rope*.

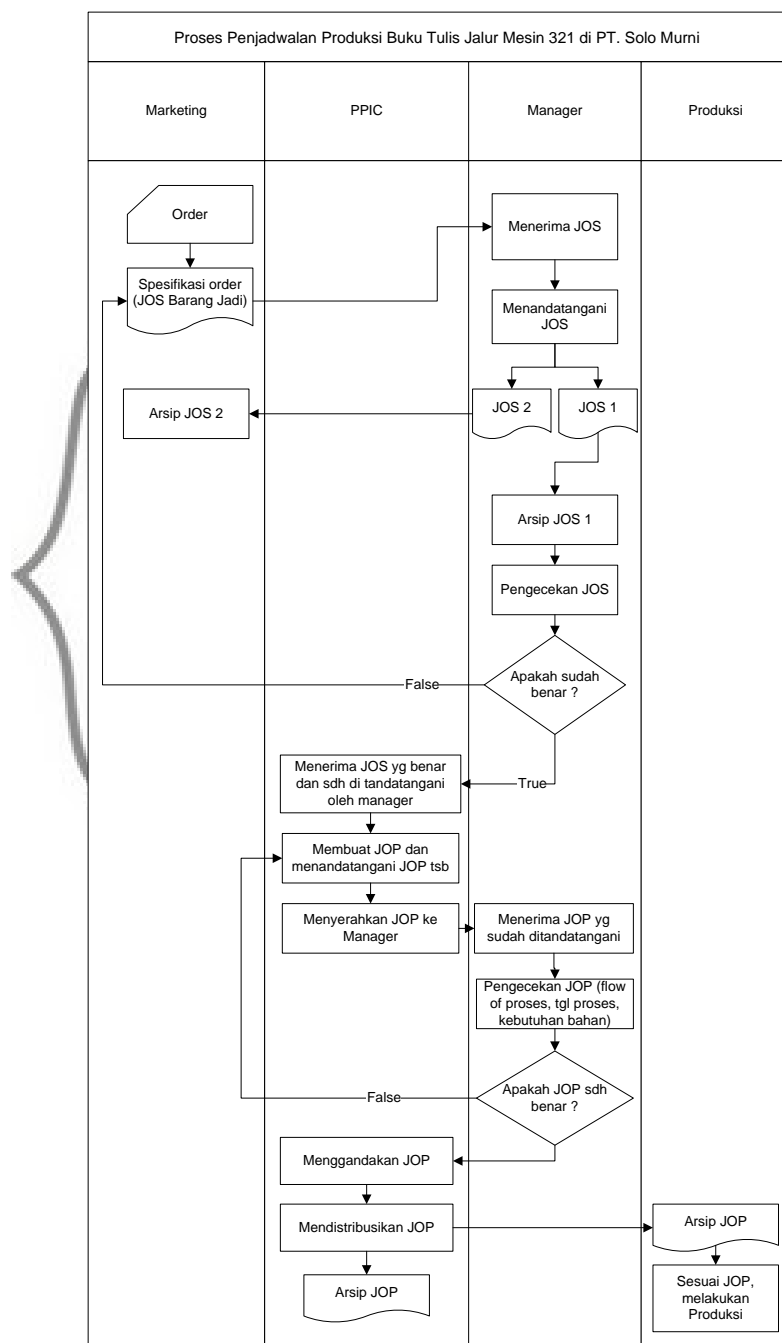
2.1 Sistem Penjadwalan Produksi di PT. Solo Murni

ISO 9001 telah diberlakukan pada PT. Solo Murni, menjelaskan bahwa jadwal produksi buku tulis digunakan sebagai pedoman di dalam melaksanakan produksi buku tulis yang sesuai dengan spesifikasi yang telah ditentukan oleh pemesan agar menghasilkan proses produksi secara cepat dan tepat (PT. Solo Murni, 2009). Perencanaan proses produksi buku tulis ini berlaku sebagai pedoman kerja bagian Departemen PPIC, bagian Produksi dan bagian Departemen Marketing. Perencanaan tersebut meliputi dari penerimaan JOS dari Departemen Marketing sampai dengan pembuatan JOP oleh Departemen PPIC dan pendistribusiannya pada bagian Produksi. JOS merupakan surat pesanan dari Departemen Marketing untuk dibuatkan barang sesuai spesifikasi yang ditujukan ke Departemen PPIC. Sedangkan JOP adalah Job Order Produksi, sebagai perintah kerja dari Departemen PPIC terhadap bagian Produksi.

Pada penjadwalan produksi buku tulis tersebut, PPIC Manager menerima JOS yang telah ditandatangani secara lengkap rangkap dua, dari pembuat JOS, yaitu JOS barang jadi oleh Marketing. Selanjutnya PPIC Manager menandatangani JOS sebagai tanda terima, dan lembar kedua dikembalikan ke pembuat JOS dan lembar kesatu tetap dibawa PPIC Manager. PPIC Manager dengan dasar JOS lembar kesatu melakukan pengecekan, apabila JOS salah maka dikembalikan ke pusat pembuat JOS untuk dilakukan koreksi, dan jika sudah benar oleh PPIC Manager diteruskan ke pembuat jadwal (*scheduller*) (PT. Solo Murni, 2009).

Pembuat jadwal (*scheduller*) menerima JOS, selanjutnya dengan dasar JOS dilanjutkan membuat JOP dan menandatangani selaku pembuat JOP, kemudian menyerahkan JOP ke PPIC Manager untuk dilakukan pengecekan. PPIC Manager tersebut melakukan pengecekan JOP yang meliputi pengecekan

terhadap *flow of process*, tanggal proses dan kebutuhan bahan. Setelah dilakukan pengecekan, PPIC Manager menandatangani JOP jika sudah benar, yang selanjutnya diperbanyak dan didistribusikan ke bagian: Pra-Produksi (Montage dan Plate), Produksi dan PPIC sebagai arsip (PT. Solo Murni, 2009). Secara sederhana, sistem penjadwalan produksi buku tulis jalur mesin 321 di PT. Solo Murni digambarkan dengan *flowchart* pada Gambar 2.1.



Gambar 2.1 Sistem penjadwalan produksi buku tulis jalur mesin 321 di PT. Solo Murni

Sumber: PT. Solo Murni, 2009

Ketika ada order dari Marketing, berikut dengan spesifikasi produk tersebut, selanjutnya dibuat JOS barang jadi oleh Marketing. JOS tersebut yang mendukung dalam penjadwalan produksi yang dilakukan PPIC hingga dihasilkan JOP sebagai pedoman dalam produksi di lantai produksi

ISO 9001 di PT. Solo Murni menjelaskan penjadwalan produksi buku tulis dengan pedoman JOS yang dilakukan oleh PPIC, dijadwalkan secara maju (*forward scheduling*). Penjadwalan produksi buku tulis tersebut dimulai dari bagian D303 untuk proses Cetak Isi, Cetak Cover, Vurnish Cover dan Embosst Cover; bagian D304 untuk proses Lipat Isi dan Potong Cover; dan bagian D305 untuk proses Mesin 321 (PT. Solo Murni, 2009).

2.2 Penjadwalan Produksi

Penjadwalan produksi didefinisikan sebagai proses pengalokasian sumber daya untuk menyelesaikan sekumpulan tugas (Baker, 1974). Penjadwalan merupakan proses pengorganisasian, pemilihan, penggunaan waktu untuk menangani aktivitas-aktivitas yang diperlukan untuk memproduksi produk tertentu pada waktu tertentu sesuai dengan jumlah waktu yang tersedia dan keterbatasan antara aktivitas dan sumber daya yang tersedia (Sipper dan Bulfin, 1997).

Proses penjadwalan produksi memerlukan tiga informasi dasar untuk setiap order (Baker, 1974), yaitu:

1. *Processing time* atau waktu proses (t_j). Jumlah waktu yang diperlukan oleh *job j*.
2. *Ready time* atau saat siap (r_j). Kondisi dimana *job j* telah tersedia untuk diproses.
3. *Due date* atau saat kirim (d_j). Kondisi dimana pemrosesan *job j* harus selesai.

Perangkat dasar yang digunakan untuk mengevaluasi penjadwalan produksi ada empat (Baker, 1974), yaitu:

1. *Completion time* (C_j). Waktu dimana pemrosesan *job j* diselesaikan. Ukuran kuantitatif untuk mengevaluasi penjadwalan biasanya adalah fungsi *completion time*.

2. *Flow time* (F_j). Jumlah waktu yang dihabiskan *job j* di dalam sistem. $F_j = C_j - r_j$. *Flow time* mengukur respon sistem pada permintaan-permintaan secara individual untuk pemrosesan dan merepresentasikan interval menunggu sebuah *job* antara kedatangan *job* dan penyelesaian *job*. Interval ini sering disebut *turnaround time*.
3. *Lateness* (L_j). Selisih waktu antara *completion time job j* dengan *due date job j*. $L_j = C_j - d_j$. *Lateness* mengukur kesesuaian penjadwalan dengan *due date*. Hal yang perlu diperhatikan adalah bahwa bisa jadi suatu *job* diselesaikan lebih awal dari *due date*, yang disebut *negative lateness*. *Negative lateness* menunjukkan bahwa pemrosesan lebih baik dari *due date* yang diharapkan, sedangkan *positive lateness* menunjukkan pemrosesan yang lebih buruk dari *due date*.
4. *Tardiness* (T_j) atau *positive lateness*. Keterlambatan *job j* jika *job j* tidak sesuai dengan *due date* atau keterlambatan nol. $T_j = \max \{0, L_j\}$. *Tardiness* (T_j) atau *positive lateness* biasanya digunakan untuk mengukur suatu keterlambatan.

Penjadwalan umumnya dievaluasi dengan menghitung secara keseluruhan (agregat) yang mengumpulkan informasi tentang semua *job*, menghasilkan ukuran performansi dalam bentuk satu dimensi. Ukuran performansi penjadwalan biasanya merupakan fungsi dari sejumlah *completion time* dalam sebuah jadwal produksi. Ukuran performansi agregat yang bisa digunakan (Baker, 1974) adalah:

1. *Mean Flow Time* : $\bar{F} = \frac{1}{n} \sum_{j=1}^n F_j$ (2.1)

2. *Mean Lateness* : $\bar{L} = \frac{1}{n} \sum_{j=1}^n L_j$ (2.2)

3. *Mean Tardiness* : $\bar{T} = \frac{1}{n} \sum_{j=1}^n T_j$ (2.3)

4. *Maximum Flow Time* : $F_{\max} = \max_{1 \leq j \leq n} \{F_j\}$ (2.4)

5. *Maximum Tardiness* : $T_{\max} = \max_{1 \leq j \leq n} \{T_j\}$ (2.5)

6. *Number of Tardy Job* : $N_T = \sum_{j=1}^n \delta(T_j)$, (2.6)

dimana $\delta(x) = 1$, jika $x > 0$

$$\delta(x) = 0, \text{ lainnya}$$

Beberapa aturan prioritas yang sering digunakan dalam penjadwalan produksi adalah (Fogarty et. al., 1991):

1. *First Come First Served (FCFS)*. Urutan pengerjaan order dilakukan berdasarkan waktu kedatangan order yang tercepat.
2. *Shortest Processing Time (SPT)*. Urutan pengerjaan order dilakukan berdasarkan waktu pemrosesan order yang terpendek.
3. *Shortest Total Processing Time Remaining (STPT)*. Urutan pengerjaan order dilakukan berdasarkan waktu sisa pemrosesan terpendek.
4. *Earliest Due Date (EDD)*. Urutan pengerjaan order dilakukan berdasarkan *due date* order yang tercepat. Aturan ini bekerja baik jika waktu proses hampir sama.
5. *Fewest Operation (FO)*. Urutan pengerjaan order dilakukan berdasarkan jumlah operasi yang paling sedikit.
6. *Slack Time (ST)*. Urutan pengerjaan order dilakukan berdasarkan *slack time* yang terkecil. $ST = \text{due date} - \text{present date} - \text{remaining processing time}$.
7. *Critical Ratio (CR)*. Urutan pengerjaan order dilakukan berdasarkan *critical ratio* yang terkecil. $CR = (\text{due date} - \text{present date}) / \text{MLT}$.

Kriteria untuk mengevaluasi sistem kontrol prioritas dapat dikategorikan sebagai berikut (Fogarty et. al., 1991):

1. Persentase ketepatan pengiriman order.
 - a. Pelanggan.
 - b. Lini perakitan.
2. Rata-rata keterlambatan (*mean tardiness*).
3. *Work-in-process (WIP)*.
4. *Idle time*.
5. Meminimasi *setup time*.
6. Penghematan energi.

2.3 Sistem *Bottleneck*

2.3.1 Prinsip-prinsip *Bottleneck* – *Optimized Production Technology*

Sistem penjadwalan *optimized production technology* (*OPT*) dikembangkan tahun 1970-an oleh Eliyahu Goldratt seorang fisikawan Israel. Goldratt terjun dalam bidang produksi karena beliau diminta membantu temannya yang memiliki pabrik kandang ayam. Goldratt memperkenalkan *OPT* di United State of America pada tahun 1979 dimulai dari Creative Output Inc. (COI). Konsep manajerial *OPT* disebut *Theory of Constraints* (*TOC*). Dasar pemikiran *TOC* adalah “*Anything that limits a system from achieving higher performance in attaining its goal*”. Tujuan *OPT* dan *TOC* adalah “*Make money in the present as well as in the future*”. Selanjutnya untuk menyempurnakan tujuan tersebut perusahaan harus secara simultan meningkatkan *throughput*, mengurangi inventori, dan memangkas biaya operasi (Narasimhan, et., al., 1995) (Sipper dan Bulfin, 1997).

Dasar pemikiran *OPT* adalah bahwa *bottlenecks* merupakan dasar untuk penjadwalan dan perencanaan kapasitas. Sumber daya dibedakan menjadi dua, yaitu sumber daya *bottleneck* (SDB) dan sumber daya *nonbottleneck* (SDNB). SDB dijadwalkan untuk memaksimalkan utilitas, sedangkan SDNB dijadwalkan untuk mendukung SDB (Sipper dan Bulfin, 1997). *OPT* bukan merupakan metode pertama yang menangani sistem *bottleneck* karena dalam manajemen proyek lintasan kritis juga telah diidentifikasi. Konsep manajemen proyek ini digabungkan dengan *OPT rules* yang diformulasikan untuk mencapai utilitas maksimum SDB. *OPT rules* yang diperkenalkan oleh Jacobs (1984) dalam Narasimhan, et., al., (1995), Sipper dan Bulfin (1997) dapat dilihat pada Tabel 2.1.

Tabel 2.1 *OPT Rules*

| No. | Rule |
|-----|--|
| 1. | Menyeimbangkan aliran, bukan kapasitas. |
| 2. | Stasiun kerja <i>bottleneck</i> (SKB) menentukan utilitas stasiun kerja <i>nonbottleneck</i> (SKNB). |
| 3. | Utilitas dan aktivasi sumber daya tidak sama. |
| 4. | Kehilangan jam kerja pada SKB merupakan kehilangan jam kerja pada sistem secara keseluruhan. |
| 5. | Jam kerja yang dijaga pada SKB merupakan suatu bayangan. |
| 6. | SKB menentukan <i>throughput</i> dan <i>inventori</i> dalam sistem. |
| 7. | Perpindahan <i>batch</i> bisa jadi atau kadang-kadang tidak sama dengan proses <i>batch</i> . |
| 8. | Proses <i>batch</i> sebaiknya berubah-ubah, tidak tetap. |
| 9. | Penjadwalan sebaiknya dibuat dengan memperhatikan keseluruhan bagian yang <i>bottleneck</i> . |
| 10. | <i>Lead time</i> merupakan hasil dari penjadwalan dan tidak bisa diprediksikan sebelumnya. |

Sumber: Narasimhan, et., al., 1995; Sipper dan Bulfin, 1997

2.3.2 *Theory of Constraints*

Theory of Constraints (TOC) merupakan hasil perkembangan dan peningkatan dari *OPT*. TOC juga sering disebut *thoughtware*, *synchronous production*, dan *synchronized manufacturing*. Dasar pemikiran TOC adalah keluaran sistem ditentukan oleh stasiun *bottleneck* sebagai konstrain. Definisi konstrain dalam TOC mempunyai aplikasi yang lebih luas di luar perencanaan dan pengendalian produksi. Konstrain tersebut seharusnya diidentifikasi dan diatur untuk meningkatkan performansi perusahaan (Narasimhan, et., al., 1995). Tiga kategori konstrain adalah (Sipper dan Bulfin, 1997):

1. *Internal resource constraint*. *Bottleneck* yang terjadi pada rantai produksi seperti mesin, pekerja, dan peralatan.
2. *Market constraint*. Kondisi ini terjadi bila permintaan pasar lebih kecil dari kapasitas produksi. Dalam kasus ini pasar menentukan jalannya produksi.
3. *Policy constraint*. Dalam kondisi ini kebijakan menentukan tingkat produksi.

TOC memfokuskan pada aturan yang diterapkan pada sistem *bottleneck* untuk memperbaiki performansi sistem dan mencapai tujuan (Narasimhan, et., al., 1995). Dua ukuran performansi yang digunakan adalah (Sipper dan Bulfin, 1997):

1. *Financial measurement*. Ukuran yang sering digunakan adalah *net profit*, *return on investment*, dan *cash flow*.
2. *Operational measurement*. Ukuran yang digunakan, yaitu:
 - a. *Throughput*. Tingkatan yang mana uang dihasilkan oleh sistem dalam bentuk penjualan.
 - b. *Inventory*. Uang yang oleh sebuah sistem diinvestasikan dalam pembelian material. Hal ini mengukur inventori hanya pada biaya material tanpa biaya tenaga kerja dan *overhead*.
 - c. *Operating expenses*. Uang yang oleh sebuah sistem digunakan untuk mengubah inventori menjadi *throughput*, termasuk biaya tenaga kerja, *overhead*, dan biaya-biaya yang lain.

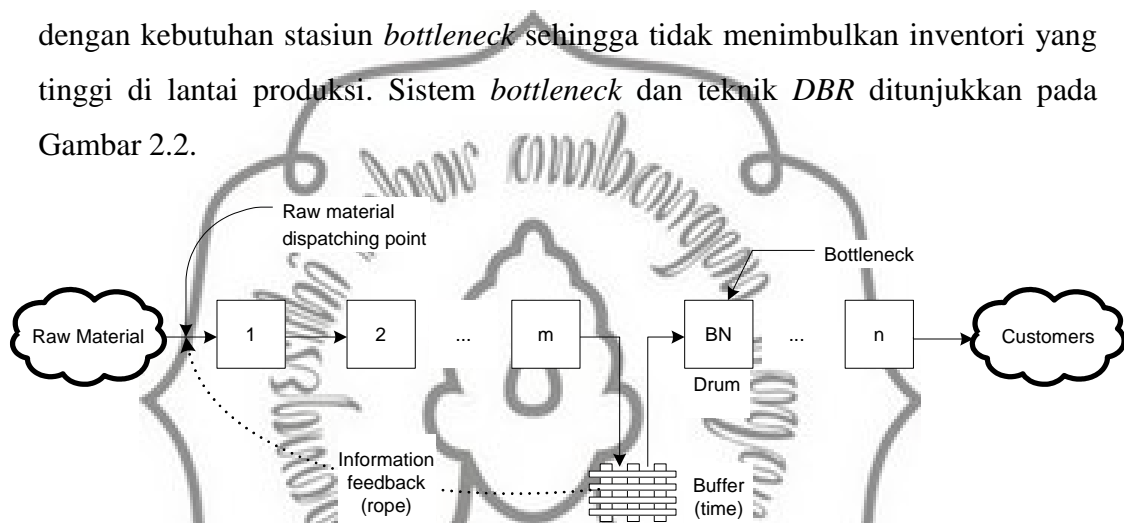
Goldratt mengembangkan lima langkah untuk memperbaiki sistem *bottleneck* secara terus menerus. Lima langkah tersebut adalah (Narasimhan, et., al., 1995):

1. Mengidentifikasi bagian yang konstrain pada sistem (*Identifying constraint*).
2. Memutuskan bagaimana mengeksplotasi bagian yang konstrain pada sistem (*Exploiting constraint*).
3. Mengsubordinasikan bagian-bagian lain yang mendukung keputusan pada langkah 2 (*Subordinate all parts on manufacturing system*).
4. Tingkatkan bagian yang konstrain pada sistem (*Elevating constraint*).
5. Jika bagian yang konstrain telah dipecahkan kembali ke langkah 1.

TOC menyarankan beberapa pendekatan yang spesifik untuk mengimplementasikan lima langkah tersebut. Teknik-teknik tersebut adalah *effect-cause-effect-analysis*, *evaporating clouds*, *buffer management*, dan *drum-buffer-rope*.

2.3.3 Teknik *Drum-Buffer-Rope*

Drum-Buffer-Rope (DBR) merupakan teknik pengendalian produksi untuk mengimplementasikan langkah-langkah dalam *TOC*, yaitu eksploitasi, subordinasi, dan elevasi (Sipper dan Bulfin, 1997). Jika suatu sistem memiliki stasiun *bottleneck* maka secara alami stasiun *bottleneck* ini menjadi titik pengendali sistem secara menyeluruh. Tingkat produksi stasiun *bottleneck* menentukan tingkat produksi sistem keseluruhan. Stasiun *bottleneck* disebut **drum** sebagai titik pengendali. Alasan utama menggunakan stasiun *bottleneck* sebagai titik pengendali adalah untuk menjamin stasiun *upstream* berproduksi sesuai dengan kebutuhan stasiun *bottleneck* sehingga tidak menimbulkan inventori yang tinggi di rantai produksi. Sistem *bottleneck* dan teknik *DBR* ditunjukkan pada Gambar 2.2.



Gambar 2.2 Sistem *Bottleneck* dan Teknik *DBR*

Sumber: Sipper dan Bulfin, 1997

Sistem *bottleneck* memiliki **buffer** yang diletakkan sebelum stasiun *bottleneck*. Tujuan utama *buffer* adalah untuk melindungi stasiun *bottleneck* dari fluktuasi dan variasi tingkat produksi. Ukuran *buffer* diukur dalam waktu standar, yaitu waktu yang dibutuhkan stasiun *bottleneck* untuk memproses semua item di stasiun *bottleneck*. Buffer dapat dikategorikan menjadi dua (Sipper dan Bulfin, 1997), yaitu:

1. *Time buffer*, yaitu waktu yang dijadikan penyangga dengan tujuan untuk melindungi laju produksi sistem dari gangguan yang sering terjadi pada sistem produksi.
2. *Stock buffer*, yaitu produk akhir atau produk antara yang dijadikan penyangga dengan tujuan memperbaiki kemampuan menanggapi sistem produksi

terhadap permintaan pasar sehingga sistem masih memungkinkan untuk menyelesaikan produk di bawah waktu penyelesaian normal.

Buffer dihubungkan dengan *raw material dispatching point* pada bagian masuk lini produksi melalui mekanisme arus balik yang disebut *rope*. *Dispatching point* hanya akan me-*release* material secukupnya sehingga inventori di lantai produksi bisa dikendalikan.

2.3.4 Penjadwalan *Bottleneck*

Berdasarkan teknik *DBR* ditentukan tiga langkah dalam penjadwalan *DBR* sebagai berikut (Sipper dan Bulfin, 1997):

1. Mengidentifikasi bagian yang *bottleneck* pada sistem

Identifikasi bagian yang *bottleneck* pada sistem dapat dilakukan dengan dua cara, yaitu dengan melihat secara langsung *work-in-process* (*WIP*) di lantai produksi dan dengan menghitung beban kerja di setiap stasiun. Stasiun kerja dengan *WIP* tertinggi dan atau beban kerja tertinggi kemudian ditetapkan sebagai stasiun kerja *bottleneck*.

2. Melakukan penjadwalan produksi pada bagian yang *bottleneck*.

Penjadwalan produksi dilakukan dengan menentukan terlebih dahulu *bottleneck release time for job i* ($r_i^{(b)}$) dan *bottleneck due date for job i* ($d_i^{(b)}$).

Termasuk di dalam $r_i^{(b)}$ adalah waktu proses dan waktu tunggu untuk operasi pada stasiun kerja *upstream*. Termasuk di dalam $d_i^{(b)}$ adalah waktu proses dan waktu tunggu untuk operasi pada stasiun kerja *downstream*. Formula di bawah ini mengasumsikan bahwa waktu tunggu = 0.

$$r_i^{(b)} = r_i + \sum_{l=1}^{j(b)-1} p_{il} \quad (2.1)$$

$$d_i^{(b)} = d_i - \sum_{l=j(b)+1}^N p_{il} \quad (2.2)$$

Dimana,

- b : Stasiun kerja *bottleneck*
 N : Jumlah stasiun kerja di lini produksi
 $j(b)$: Operasi job i yang dilakukan di b

- p_{il} : Waktu proses job i pada stasiun kerja l
 r_i : Waktu *release* job i
 d_i : *Due date* job i
 $r_i^{(b)}$: Waktu *release* job i di b
 $d_i^{(b)}$: *Due date* job i di b

3. Melakukan penjadwalan produksi pada stasiun *upstream* dan *downstream*.

Penjadwalan produksi pada stasiun kerja *upstream* dilakukan dengan pendekatan *backward*, sedangkan pada stasiun kerja *downstream* dilakukan dengan pendekatan *forward*.

2.4 Manajemen *Lead Time*

Lead time adalah selang waktu yang dibutuhkan untuk menyelesaikan aktifitas. Komponen-komponen *lead time* antara lain: waktu persiapan order, *queue time*, waktu transportasi, waktu inspeksi dan penerimaan. *Lead time* tersebut perlu dimanajemen sehingga efisiensi waktu menjadi lebih baik. Dalam memenejemen *lead time* digunakan algoritma Zijm-Buitenhek yang merupakan suatu algoritma untuk memperkirakan *mean waiting time* dan *mean lead time* di setiap stasiun kerja dengan pendekatan teori antrian.

Simbol-simbol:

- n : Jumlah order i
 i : Order i ($i = 1, 2, \dots, n$)
 N : Jumlah stasiun kerja di lini produksi
 j : Stasiun kerja j ($j = 1, 2, \dots, N$)
 p_{ij} : Waktu proses order i di stasiun kerja j
 τ_{ij} : Waktu setup operasi k order i di stasiun kerja j
 a_{ij} : Waktu proses per unit order i di stasiun kerja j
 Q_{ij} : Ukuran *lot* order i
 D_{ij} : *Demand rate* order i
 δ_{ij} : 1, jika order i dikerjakan di stasiun kerja j
 0, jika yang lainnya
 λ_{ij} : *Arrival rate* order i stasiun kerja j

λ_j : *Arrival rate* stasiun kerja j

ρ_j : *Average work load* di stasiun kerja j

W_j : *Mean waiting time* di stasiun kerja j

ET_{ij} : *Mean lead time* order i di stasiun kerja j

ET_i : *Mean shop manufacturing lead time* order i

Ep_j : Waktu proses operasi-independent di stasiun kerja j

Persamaan-persamaan yang digunakan dalam algoritma Zijm-Buitenhek dalam Utomo (2005) adalah sebagai berikut:

1. Perhitungan waktu proses order i di stasiun kerja j (p_{ij})

$$p_{ij} = \tau_{ij} + Q_i \cdot a_{ij} \quad (2.3)$$

2. Perhitungan *arrival rate* order i di stasiun kerja j (λ_{ij})

$$\lambda_{ij} = \frac{D_i}{Q_i} \cdot \delta_{ij} \quad (2.4)$$

3. Perhitungan *average work load* di stasiun kerja j (ρ_j)

$$Ep_j = \frac{\sum_{i=1}^n \lambda_{ij}}{\sum_{i=1}^n \lambda_j} \cdot (p_{ij}) = \sum_{i=1}^n \frac{\lambda_{ij}}{\lambda_j} \cdot (p_{ij}) \quad (2.5)$$

$$\rho_j = \lambda_j \cdot E(p_j) = \lambda_j \cdot \sum_{i=1}^n \frac{\lambda_{ij}}{\lambda_j} \cdot (p_{ij}) = \sum_{i=1}^n \lambda_{ij} \cdot p_{ij} \quad (2.6)$$

4. Perhitungan *mean waiting time* di stasiun kerja j (W_j)

$$EW_j = \frac{\lambda_j \cdot E(p_j)^2}{2(1 - \rho_j)} = \frac{\lambda_j \cdot \sum_{i=1}^n \frac{\lambda_{ij}}{\lambda_j} \cdot (p_{ij})^2}{2(1 - \rho_j)} = \frac{\sum_{i=1}^n \lambda_{ij} \cdot (p_{ij})^2}{2(1 - \rho_j)} \quad (2.7)$$

5. Perhitungan *mean lead time* order i di stasiun kerja j (ET_{ij})

$$ET_{ij} = EW_j + p_{ij} \quad (2.8)$$

6. Perhitungan *mean shop manufacturing lead time order i* (ET_i)

$$ET_i = \sum_{j=1}^N \delta_{ij} \cdot ET_{ij} \quad (2.9)$$

2.5 Pemrograman Komputer

2.5.1 Algoritma

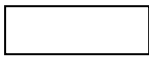
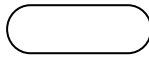
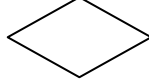
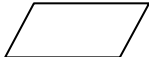




Dalam Iftadi (2006) menjelaskan bahwa kata algoritma ditemukan oleh ahli matematika dari Uzbekistan yang bernama Abu Abdullah Ibn Musa al-Khawrizmi (770 – 840M). kata ini kemudian digunakan untuk menyebut konsep algoritma yang ditemukannya. Banyak cabang ilmu komputer yang diacu dari terminology algoritma, tetapi tidak berarti algoritma selalu identik dengan ilmu komputer. Algoritma dapat disusun dalam bahasa natural (bahasa Indonesia, bahasa Inggris, dll), menggunakan diagram alir (*flow chart*) dan menggunakan *pseudo-code*.

Menurut Kamus Besar Bahasa Indonesia, algoritma adalah urutan logis pengambilan keputusan untuk pemecahan masalah. Algoritma adalah urutan langkah-langkah logis penyelesaian masalah yang disusun secara sistematis (Iftadi, 2006). Dari definisi tersebut, logis disini berarti kebenarannya harus dapat ditentukan benar dan salahnya.

Langkah-langkah dalam penyelesaian masalah pada algoritma tersebut dapat berupa runtutan tindakan, pemilihan alternative tindakan dan pengulangan (Iftadi, 2006). Runtutan terdiri atas satu atau lebih perintah dan tiap-tiap perintah akan dikerjakan secara berurutan. Pemilihan adalah pengerjaan perintah jika memenuhi persyaratan tertentu. Sedangkan pengulangan adalah pengerjaan perintah (kelompok kegiatan).

2.5.2 Diagram Alir

Diagram alir merupakan diagram yang menggambarkan urutan logika proses pemecahan masalah. Pada diagram alir penyelesaian masalah dilakukan dengan diwakili oleh simbol-simbol tertentu sehingga lebih mudah dimengerti. Simbol-simbol tersebut tergambar pada gambar 2.3 dibawah ini:

| | |
|---|---|
|  | Kotak segiempat digunakan untuk menyatakan suatu proses dan pernyataan. |
|  | Oval menyatakan awal atau akhir program. |
|  | Belah ketupat digunakan untuk menyatakan pengujian kondisi syarat, keluarannya adalah <i>true</i> atau <i>false</i> . |
|  | Trapesium menyatakan permintaan input, penampilan keluaran melalui alat I/O |
|  | Anak panah menunjukkan arah proses, sesuai dengan arah yang ditunjukkan. |
|  | Kotak segienam mewakili proses inisiasi data beik variable ataupun konstanta. |
|  | Menyatakan atau mewakili suatu proses yang terdefinisi. |
|  | Penghubung antar terminal sesuai indeks atau label yang dikenakan |

Gambar 2.3 Simbol-simbol diagram alir

Sumber: Iftadi, 2006

2.6 Program Aplikasi

Program aplikasi sekarang telah berkembang sedemikian luas seiring dengan semakin menjamurnya *personal computer* dan *laptop*. Aplikasi menjadi fitur yang wajib ada dalam setiap komputer.

2.6.1 Definisi Program Aplikasi

Program aplikasi (*application program*) adalah bagian dari perangkat lunak komputer yang memiliki kemampuan dari komputer secara langsung dan menyeluruh untuk sebuah tugas dimana pengguna aplikasi mengharapkan untuk melaksanakan. Perangkat lunak tersebut merupakan program komputer yang berfungsi sebagai sarana interaksi antara pengguna dan perangkat keras. Perangkat lunak ini dapat juga disebut sebagai penterjemah perintah-perintah yang dijalankan pengguna komputer untuk diteruskan atau diproses oleh perangkat keras. Perangkat lunak inidibagi menjadi tiga tingkatan (www.wikipedia.com, 2010), yaitu:

1. Tingkatan program aplikasi, misalnya *Microsoft office*.

2. Tingkatan sistem operasi, misalnya *Microsoft Windows*.
3. Tingkatan bahasa pemrograman, yang dibagi atas bahasa pemrograman tingkat tinggi seperti Pascal dan bahasa pemrograman tingkat rendah yaitu bahasa *assembly*.

Dalam ilmu komputer, aplikasi adalah program komputer yang dirancang sedemikian rupa untuk membantu pengguna dalam menyelesaikan suatu tipe pekerjaan (www.wikipedia.com, 2010). Suatu aplikasi berbeda dengan sistem operasi yang menjalankan komputer, *utility* yang melakukan pemeliharaan dan bahasa pemrograman yang membuat program komputer. Berdasarkan pada pekerjaan yang telah dirancang, suatu aplikasi dapat memanipulasi teks, angka, grafik, maupun kombinasi dengan elemen tersebut. Beberapa aplikasi menawarkan kekuatan komputasi dengan memfokuskan diri pada pekerjaan tunggal, seperti memproses kata, mengolah grafik ataupun memanipulasi angka. Contoh aplikasi tersebut yaitu: *Abiword*, *Blender 3D*, *GIMP* dan lain-lain. Beberapa aplikasi lain disebut perangkat lunak integrasi menawarkan kekuatan yang memiliki beberapa aplikasi didalamnya seperti *word processor*, *spreadsheet* ataupun program *database*. Contoh dari aplikasi tersebut misalnya *Microsoft Office*, *Netbeans* dan lain-lain.

2.6.2 Klasifikasi Program Aplikasi

Setelah munculnya *personal computer* (PC), maka kebutuhan akan program aplikasi semakin meningkat sehingga banyak jenis program aplikasi bermunculan. Secara umum, ada delapan tipe program aplikasi yang ada dan berkembang (www.wikipedia.com, 2010), antara lain:

1. *Enterprise software*

Enterprise software muncul berdasarkan kebutuhan organisasi akan proses organisasi serta aliran data skala luas.

2. *Enterprise infrastructure software*

Enterprise infrastructure software mendukung kemampuan dalam system *enterprise software*.

3. *Information worker software*

Information worker software muncul didasarkan atas kebutuhan individu untuk menciptakan dan mengatur informasi. *Content acces software* digunakan untuk mengakses sebuah *content* tanpa merubahnya, tetapi adapula beberapa aplikasi yang menyediakan fitur untuk merubah *content*.

4. *Educational software*

Educational software ini berhubungan dengan fungsi media dan hiburan, tetapi memiliki tujuan khusus untuk pendidikan.

5. *Simulation software*

Simulation software digunakan untuk menyimulasikan system fisik atau abstrak untuk keperluan penelitian, pelatihan ataupun hiburan.

6. *Media development software*

Media development software ini muncul karena kebutuhan individu dalam menghasilkan karya elektronik untuk keperluan komersial.

7. *Product engineering software.*

Product engineering software digunakan untuk mengembangkan *hardware* dan produk *software*. Contohnya yaitu *Computer Aided Design (CAD)*, *Computer Aided Engineering (CAE)* dan bahasa pemrograman terintegrasi.

2.6.3 Visual FoxPro

Visual FoxPro adalah bahasa program berorientasi objek dan prosedural dari Microsoft (www.wikipedia.com, 2010). Awalnya bahasa pemrograman ini dikenal dengan nama FoxBASE yang diluncurkan oleh Fox Software pada awal 1984. Fox Technologies kemudian bergabung dengan Microsoft pada 1992 sehingga di depan nama FoxBASE ditambahkan awalan "Visual".

Versi terakhir FoxPro (9.0) dapat berjalan pada sistem operasi Mac OS, DOS, Windows, dan Unix. Visual FoxPro 3.0, versi "Visual" pertama, akhirnya tersingkir karena hanya mendukung Mac OS dan Windows, dan versi berikutnya hanya mendukung Windows saja. Versi terkini Visual FoxPro adalah berdasarkan teknologi COM dan Microsoft telah menyatakan bahawa mereka tidak berniat untuk mencipta versi Microsoft .NET.

FoxPro berasal dari anggota kelas bahasa umum yang mengacu pada bahasa “xBase”, yang mempunyai sintak berdasar pada bahasa program dBase (www.wikipedia.com, 2010). Anggota lain famili bahasa xBase meliputi Clipper dan Recital. Visual FoxPro (VFP) sulit berhubungan dengan mesin *database* relationalnya, yang memberikan kemampuan xBase FoxPro untuk membantu SQL *query* dan manipulasi data. Tidak seperti sistem manajemen *database*, VFP banyak keistimewaan, bahasa program dinamik yang tidak membutuhkan penambahan tujuan umum pemrograman. Ini dapat digunakan menulis aplikasi “*fat client*”, tapi juga *middleware* dan aplikasi web.

Pada gambar 2.4 dibawah ini menjelaskan mengenai kecocokan sistem operasi dari masing-masing versi Visual FoxPro dengan masing-masing versi Windows, adalah sebagai berikut (www.wikipedia.com, 2010):

| Version | VFP 3.0 | VFP 5.0 | VFP 6.0 | VFP 7.0 | VFP 8.0 | VFP 9.0 |
|------------------------|--------------------|---------|---------|--------------|--------------|--------------|
| Windows 3.x | Yes | No | No | No | No | No |
| Windows NT 4.0 | Yes | Yes | Yes | Yes | No | No |
| Windows 95 | Yes | Yes | Yes | Runtime only | No | No |
| Windows 98 | Yes | Yes | Yes | Yes | Runtime only | Runtime only |
| Windows Me | Yes | Yes | Yes | Yes | Runtime only | Runtime only |
| Windows 2000 | Yes | Yes | Yes | Yes | Yes | Yes |
| Windows XP | Yes | Yes | Yes | Yes | Yes | Yes |
| Windows Server 2003 | ? | ? | Yes | Yes | Yes | Yes |
| Windows Vista | Compatibility Mode | Yes | Yes | Yes | Yes | Yes |
| Windows 7 (Build 7600) | ? | No | Yes | Yes | Yes | Yes |

Gambar 2.4 Kecocokan sistem operasi VFP dan Windows

Sumber: www.wikipedia.com

2.7 Penelitian Pendukung

Pada kajian pustaka penjadwalan produksi dengan menggunakan pendekatan *DBR* ini mengkaji dari jurnal penelitian oleh Inten Tejaasih dan Suzwamela pada Agustus 2003, dengan judul ” Aplikasi Model Penjadwalan *Drum Buffer Rope* Pada Sistem Manufaktur MTO Repetitif Dengan Sistem Produksi *Mixed Flow Shop*”.

Jurnal penelitian ini membahas mengenai pengembangan model penjadwalan *DBR* pada MTO Repetitif dengan beberapa modifikasi agar sesuai dengan kondisi yang ada di lapangan, Departemen CTV, Divisi Auto Insert PT. Sharp Yasonta Indonesia. Sistem produksi disana yaitu bersifat *mixed flow shop*, beberapa stasiun kerja yang mempunyai mesin paralel dan jumlah mesin di stasiun konstrain lebih banyak dari stasiun kerja non-konstrain. Berdasarkan hal tersebut, maka tujuan penelitian pada jurnal ini adalah membuat pengembangan model

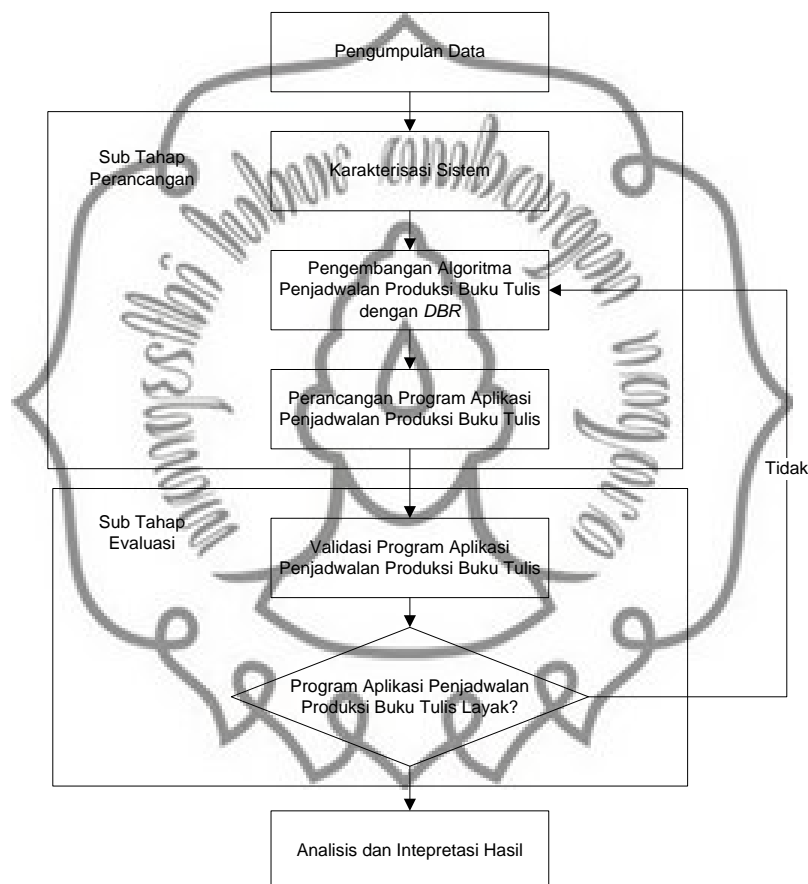
penjadwalan *DBR MTO* yang disesuaikan dengan kondisi perusahaan, untuk meminimasi rata-rata lamanya waktu yang dihabiskan oleh sebuah order di lantai produksi, yang diukur dengan *mean flowtime* dan *total actual flowtime*.

Pemecahan masalah dimulai dengan menganalisis karakteristik lantai produksi dan kemudian dilanjutkan dengan pengembangan model *DBR MTO* yang telah dikembangkan pada penelitian sebelumnya. Karena model penjadwalan *DBR MTO* yang sudah ada dibatasi pada sistem produksi *flow shop*, maka pengembangan model dilakukan agar sesuai dengan kondisi *mixed flow shop*. Selanjutnya hasil pengembangan model diuji dengan memakai data yang terdapat pada penelitian sebelumnya tersebut. Hasil pengujiannya adalah valid. Lalu, untuk mempermudah proses perhitungan dilakukan perancangan program sesuai dengan algoritma hasil pengembangan. Tahap terakhir dari penelitian ini adalah implementasi model dengan menggunakan data aktual perusahaan. Hasil implementasi menunjukkan bahwa penerapan model penjadwalan *DBR MTO* dapat mengurangi *mean flowtime* dan *total actual flowtime* pengerjaan order di lantai produksi perusahaan.



BAB III METODOLOGI PENELITIAN

Pada bab ini akan dibahas mengenai metodologi penelitian, yaitu tahapan-tahapan yang dimulai dari perumusan masalah sampai dengan kesimpulan, yang membentuk sebuah alur yang sistematis. Metodologi penelitian ini digunakan sebagai pedoman dalam melaksanakan penelitian ini agar hasil yang dicapai tidak menyimpang dari tujuan yang telah ditentukan sebelumnya. Alur metodologi penelitian bisa dilihat pada gambar 3.1.



Gambar 3.1 Langkah-langkah Penelitian

3.1 Pengumpulan Data

Pada tahap ini dilakukan pengumpulan data dan informasi yang dibutuhkan dalam penelitian. Data dan informasi diperoleh melalui wawancara langsung dengan pihak yang berwenang dan pengamatan langsung. Data yang

diperlukan untuk penjadwalan produksi buku tulis pada jalur mesin 321 dengan menggunakan pendekatan *Drum-Buffer-Rope* adalah:

- a. Data kapasitas alokasi mesin produksi buku tulis pada jalur mesin 321 untuk mengidentifikasikan bagian yang *bottleneck* (SKB).
- b. Jumlah mesin (*server*) setiap stasiun kerja pada lini produksi buku tulis pada jalur mesin 321.
- c. Jam kerja regular.
- d. Order periode bulan April sampai dengan Mei tahun 2009.
- e. Tahapan pengerjaan order.
- f. Waktu proses buku tulis.
- g. Saat siap mesin (*server*) di stasiun kerja *bottleneck*.

3.2 Karakterisasi Sistem

Pada tahap ini dilakukan karakterisasi sistem penjadwalan produksi buku tulis di jalur mesin 321 di PT. Solo Murni. Karakteristik sistem dapat dipandang sebagai suatu proses simplifikasi dan idealisasi (Murthy et. al., 1990). Dengan karakterisasi sistem ini maka sistem nyata penjadwalan produksi buku tulis dapat teridentifikasi. Sehingga dengan karakterisasi sistem ini tujuan dari permasalahan yang ada di sistem penjadwalan produksi buku tulis tercapai. karakterisasi sistem penjadwalan produksi buku tulis ini menjelaskan mengenai sistem manufaktur, sistem produksi, proses produksi, sistem *bottleneck*, dan penjadwalan produksi buku tulis pada jalur mesin 321 yang akan dirancang di PT. Solo Murni.

3.3 Perancangan Penjadwalan produksi buku tulis

Perancangan penjadwalan produksi buku tulis pada jalur mesin 321 di PT. Solo Murni dilakukan dengan pendekatan *Drum-Buffer-Rope* (DBR) dengan kriteria minimasi rata-rata *flowtime* order (*mean flowtime*). Pada perancangan penjadwalan produksi buku tulis sebagai pemecahan masalah diharapkan mampu meningkatkan performansi sistem produksi buku tulis yang melewati jalur mesin 321 di PT. Solo Murni dengan minimasi *bottleneck* (Vurnish Cover) yang terjadi pada produk *school supplies*.

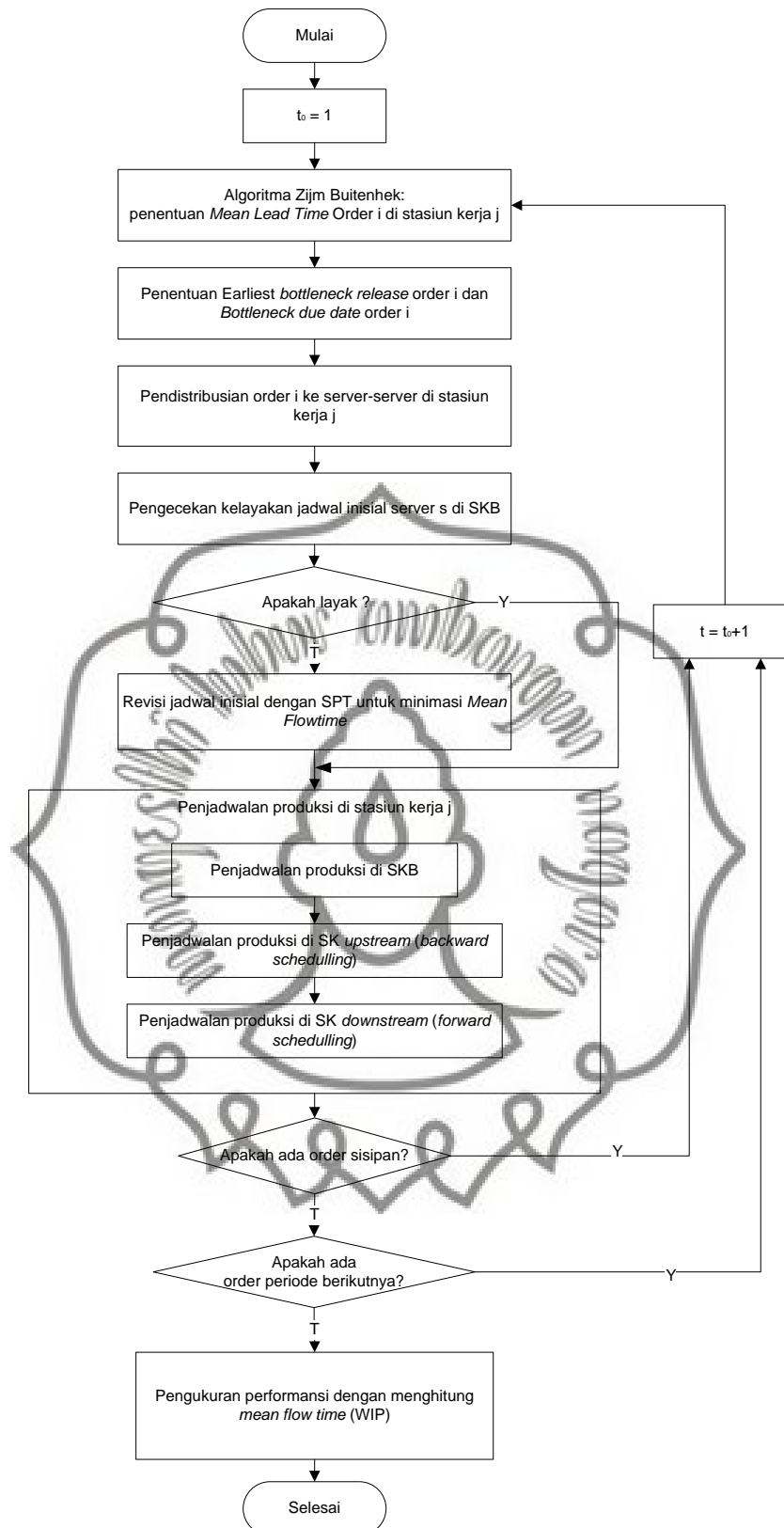
Perancangan penjadwalan produksi buku tulis yang minimasi *bottleneck* ini dikembangkan dari model *bottleneck scheduling* yang telah dilakukan Sipper

dan Bulfin (1997). Model dasar tersebut mengasumsikan tidak ada *waiting time* order di setiap stasiun kerja sehingga *lead time* order sebelum SKB merupakan jumlah waktu proses order di setiap stasiun kerja sebelum SKB, sedangkan *lead time* order sesudah SKB merupakan jumlah waktu proses order di setiap stasiun kerja setelah SKB.

Pada dasarnya model merupakan gambaran dari sistem keadaan nyata. Sedangkan pemodelan sistem merupakan proses membangun atau membentuk suatu model dari sistem nyata berdasarkan sudut pandang tertentu (Murthy et. al., 1990). Suatu model tersebut memberikan kemampuan dalam pengambil keputusan untuk menganalisis masalah secara penuh melalui pengembangan dan perbandingan alternatif keputusan. Jadi, dengan perancangan penjadwalan produksi buku tulis yang dikembangkan dari model *bottleneck scheduling* ini diharapkan dapat membantu pemecahan masalah penjadwalan produksi buku tulis di jalur mesin 321.

Penentuan jadwal produksi buku tulis yang melewati jalur mesin 321 tergambar jelas pada gambar 3.2 dengan melalui beberapa tahapan. Pada tahapan penentuan *lead time* order, perhitungan *lead time* order sebelum dan sesudah SKB dilakukan dengan Algoritma Zijm-Buitenhek (Utomo, 2005). Penentuan *lead time* order sebelum dan sesudah SKB ini dilakukan beberapa tahapan, yaitu: terlebih dahulu menentukan *arrival rate* order, lalu menentukan *average workload* order, selanjutnya menentukan *mean waiting time* dan yang terakhir menghitung *mean lead time* order. *Lead time* order sebelum dan sesudah SKB digunakan untuk mengetahui selang waktu yang dibutuhkan untuk menyelesaikan aktifitas produksi buku tulis di jalur mesin 321.

Selanjutnya dilakukan penentuan *earliest bottleneck order i* dan *bottleneck due date order i*. *Earliest bottleneck order i* merupakan variabel yang mendeskripsikan saat tercepat order *i* bisa mulai diproses di SKB dan *bottleneck due date order i* merupakan variabel yang mendeskripsikan saat paling akhir order *i* harus selesai diproses SKB. Jadi kedua variabel tersebut merupakan waktu dimana kapan saat tercepat dan saat paling akhir suatu order buku tulis dapat dikerjakan untuk diproduksi di SKB (Vurnish Cover).



Gambar 3.2 Perancangan Penjadwalan produksi buku tulis Dengan Pendekatan DBR (Kriteria Minimasi *Mean Flowtime*)

Setelah itu dilakukan tahap pendistribusian order i ke server-server di SKB dengan tujuan agar order-order tersebut didistribusikan merata ke server-server yang ada atau sesuai alokasinya dan sehingga tidak bertabrakan antar order yang didistribusikan. Setelah selesai didistribusikan, selanjutnya dilakukan pengecekan jadwal inisial apakah jadwal tersebut layak atau tidak. Jika tidak layak maka jadwal akan direvisi dengan menggunakan SPT (*Shortest Processing Time*) untuk meminimasi *mean flowtime*. Minimasi *mean flowtime* dilakukan dengan SPT *sequencing* ($t_1 \leq t_2 \leq \dots \leq t_n$) dalam Baker (1974). Kelayakan jadwal inisial ini jika waktu selesai suatu order diproduksi di stasiun kerja sesuai urutan distribusi ini kurang atau sama dengan nilai *bottleneck due date* yang merupakan nilai saat paling akhir suatu order untuk dapat dikerjakan.

Tahap selanjutnya dilakukan proses penjadwalan produksi buku tulis dengan pendekatan *DBR* dimana konsentrasi atau titik kontrolnya pada stasiun kerja *bottleneck*. Setelah jadwal di SKB telah ditentukan, selanjutnya dilakukan penjadwalan di stasiun kerja *upstream* dengan menggunakan *backward scheduling* dan stasiun kerja *downstream* dengan *forward scheduling* untuk mendukung stasiun kerja *bottleneck*.

Tahap selanjutnya dilakukan pengecekan apakah terdapat order sisipan dan atau order periode berikutnya. Order sisipan merupakan order yang bersifat mendesak ketika jadwal produksi sudah beredar atau periode produksi sudah berlangsung. Ketentuan jika terdapat order sisipan pada sistem penjadwalan, maka order sisipan dikerjakan atau diproduksi pada periode minggu berikutnya. Order ini dihitung kembali mulai tahap penentuan *leadtime* dengan Algoritma Zijm-Buitenhok (Utomo, 2005) hingga tahap penjadwalan produksi. Selanjutnya tahap terakhir yang dilakukan adalah pengukuran performansi yaitu dengan meminimasi *mean flowtime*. Baker (1974) menyatakan bahwa meminimasi *mean flowtime* juga akan meminimasi *in-process-inventory* atau *work-in-process* (WIP).

3.4 Perancangan Program Aplikasi Penjadwalan produksi buku tulis

Tahap ini dilakukan perancangan program aplikasi penjadwalan produksi buku tulis di jalur mesin 321 dengan pendekatan *DBR*. Pada perancangan tersebut dilakukan dengan membuat konstruksi program. Konstruksi program tersebut meliputi diagram alir program, *syntax* program dan *interface* program yang akan

dirancang. Perancangan program aplikasi ini menggunakan *software* Visual Foxpro 8.0.

3.5 Validasi Program Penjadwalan produksi buku tulis

Validasi digunakan untuk mengetahui perbandingan antara hasil penjadwalan produksi buku tulis dengan pendekatan *DBR* yang di simulasikan dengan perhitungan manual dan hasil penjadwalan produksi buku tulis dengan pendekatan *DBR* yang di simulasikan dengan program aplikasi yang telah disusun. Jika hasil penjadwalan produksi buku tulis tersebut setelah diperbandingkan sama, maka program aplikasi hasil rancangan program yang dibuat dapat digunakan.

Tetapi sebelumnya juga dilakukan validasi terhadap algoritma penjadwalan produksi buku tulis yang melewati jalur mesin 321 dengan menggunakan *DBR*. Validasi algoritma ini dengan menggunakan kriteria performansi (Costatine dan Ulvila, 1990) yaitu dengan penilaian (*judgement*). Atribut yang dinilai pada *judgment* ini adalah kualitas dari alasan, dimana penilaian pengguna terhadap kesesuaian jawaban yang diberikan oleh sistem terhadap permasalahan yang diberikan. Penilaian tersebut dilakukan oleh orang ahli dalam penjadwalan produksi dari PT. Solo Murni. Kriteria validasi dari *Judgment* (penilaian), yaitu: Apakah algoritma penjadwalan produksi buku tulis dengan menggunakan *DBR* dapat diterapkan di PT. Solo Murni?

3.6 Analisis dan Intepretasi hasil

Menguraikan analisis dan pembahasan masalah berdasarkan metodologi penelitian yang telah dirumuskan serta menginterpretasikan hasil penelitian. Kemudian intepretasi hasil penelitian tersebut dipakai untuk membantu penarikan kesimpulan pada tahap berikutnya.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Pengumpulan Data

Pada tahap ini dilakukan pengumpulan data yang diperlukan dalam merancang suatu program aplikasi penjadwalan produksi untuk membantu penjadwalan produksi buku tulis pada jalur mesin 321 dengan menggunakan pendekatan *Drum-Buffer-Rope* adalah:

1. Data kapasitas alokasi mesin produksi buku tulis pada jalur mesin 321.
2. Jumlah mesin (*server*) setiap stasiun kerja di lini produksi buku tulis pada jalur mesin 321.
3. Jam kerja regular.
4. Order periode bulan April sampai dengan Mei tahun 2009.
5. Data status order yang dikerjakan di masing-masing stasiun kerja pada jalur mesin 321.
6. Waktu proses buku tulis.
7. Saat siap mesin (*server*) di stasiun kerja *bottleneck* (stasiun Vurnish Cover).

4.1.1 Data kapasitas alokasi mesin produksi buku tulis pada jalur mesin 321

Data kapasitas mesin produksi buku tulis pada jalur mesin 321 ini diperlukan untuk mengidentifikasi stasiun kerja *bottleneck* (SKB), stasiun kerja yang memiliki kapasitas yang tidak mencukupi (terendah) untuk produksi buku tulis yang disebut dengan stasiun konstrain, sehingga nantinya dapat dianalisa lebih lanjut dalam pengaturan kecepatan aliran produksinya. Adapun data kapasitas mesin terdapat pada tabel 4.1 tersebut dibawah ini.

Tabel 4.1 Kapasitas Mesin Produksi Buku Tulis Pada Jalur Mesin 321 (per hari)

| No | Mesin | Cover | | | | Isi | | Mesin 321 |
|----------------------------------|---------------------------------|---------------|---------------|------------------|----------------|----------------|-----------------|----------------|
| | | Cetak | Vurnish | Embost / Glitter | Potong Cover | Cetak Isi | Mesin Lipat Isi | |
| 1 | MESIN CETAK 6W2AB | 68,500 | | | | | | |
| 2 | MESIN CETAK 4W3 | | | | | 55,000 | | |
| 3 | MESIN CETAK 5W1 | | | | | 110,000 | | |
| 4 | MESIN CETAK 5W3 | | | | | 55,000 | | |
| 5 | MESIN CETAK 6W1 | | | | | 120,000 | | |
| 6 | MESIN CETAK 8W | | | | | 120,000 | | |
| 7 | MESIN ZP | | | | | 120,000 | | |
| 8 | MESIN LIPAT RF 66E | | | | | | 98,000 | |
| 9 | MESIN LIPAT RF 78A | | | | | | 98,000 | |
| 10 | MESIN LIPAT RF 78B | | | | | | 112,500 | |
| 11 | MESIN LIPAT RF 78C | | | | | | 126,000 | |
| 12 | MESIN LIPAT RF 78D | | | | | | 126,000 | |
| 13 | MESIN 321 A | | | | | | | 90,000 |
| 14 | MESIN 321 B | | | | | | | 105,000 |
| 15 | MESIN 335 | | | | | | | 90,000 |
| 16 | GLITTER | | | 20,300 | | | | |
| 17 | EMBOST RRC 2 | | | 29,750 | | | | |
| 18 | EMBOST RRC 3 | | | 29,750 | | | | |
| 19 | MESIN UV VERNISH STEINMAN AG | | 30,500 | | | | | |
| 20 | MS POT.POLAR ELECTROMAT 115-MNL | | | | 218,400 | | | |
| 21 | MS POT.POLARELECTROMAT 72-MNL | | | | 218,400 | | | |
| TOTAL (lembar plano/hari) | | 68,500 | 30,500 | 79,800 | 436,800 | 580,000 | 560,500 | - |
| TOTAL (pieces/hari) | | - | - | - | - | - | - | 285,000 |

Sumber: PT.Solo Murni, 2009

4.1.2 Jumlah *server* setiap stasiun kerja pada lini produksi buku tulis pada jalur mesin 321

Data jumlah *server* setiap stasiun kerja lini produksi buku tulis digunakan untuk memperhitungkan kapasitas produksi stasiun kerja j sehingga beban kerja untuk setiap *server* bisa diperhitungkan. *Server* di stasiun kerja bisa berupa mesin atau tim kerja sesuai dengan proses di stasiun kerja yang bersangkutan. Jumlah *server* di setiap stasiun kerja pada jalur mesin 321 ditunjukkan pada Tabel 4.2.

Tabel 4.2. Jumlah *Server* Setiap Stasiun Kerja Buku Tulis

| j | Bagian | Stasiun Kerja j | Jumlah <i>Server</i> (m_j) |
|-----|-----------------------------|----------------------------|--------------------------------|
| 1 | Cetak | Cetak Cover | 1 |
| 2 | | Vurnish Cover | 1 |
| 3 | | Emboss Cover | 3 |
| 4 | | Cetak Isi | 6 |
| 5 | Potong dan Lipat | Potong Cover | 2 |
| 6 | | Lipat Isi | 5 |
| 7 | Jahit Kawat dan Potong Jadi | Finishing dengan mesin 321 | 3 |

Sumber: PT.Solo Murni, 2009

4.1.3 Jam kerja regular

Data jam kerja regular digunakan untuk memperhitungkan kapasitas produksi selama periode penjadwalan. Data jam kerja regular ditunjukkan pada Tabel 4.3.

Tabel 4.3 Jam Kerja Regular

| Hari | Jam Kerja | | | Jam Istirahat per Shif | Jumlah Jam Kerja (jam/hari) |
|------------------------------|---------------|---------------|---------------|------------------------|-----------------------------|
| | Shif Pagi | Shif Sore | Shif Malam | | |
| Senin | 07.00 - 15.00 | 15.00 - 23.00 | 23.00 - 07.00 | 1 | 21 |
| Selasa | 07.00 - 15.00 | 15.00 - 23.00 | 23.00 - 07.00 | 1 | 21 |
| Rabu | 07.00 - 15.00 | 15.00 - 23.00 | 23.00 - 07.00 | 1 | 21 |
| Kamis | 07.00 - 15.00 | 15.00 - 23.00 | 23.00 - 07.00 | 1 | 21 |
| Jumat | 07.00 - 15.00 | 15.00 - 23.00 | 23.00 - 07.00 | 1 | 21 |
| Sabtu | 07.00 - 15.00 | 15.00 - 23.00 | 23.00 - 07.00 | 1 | 21 |
| Total Jam Kerja (jam/minggu) | | | | | 126 |

Sumber: PT.Solo Murni, 2009

4.1.4 Order buku tulis periode bulan April sampai dengan Mei tahun 2009

Data order buku tulis merupakan data order-order buku tulis yang akan dijadwalkan dalam penelitian ini. Adapun data order buku tulis ini dapat dilihat pada lampiran.

4.1.5 Data status order yang dikerjakan di masing-masing stasiun kerja pada jalur mesin 321

Status order yang dikerjakan di masing-masing stasiun kerja pada jalur mesin 321 merupakan aliran pemrosesan order di lantai produksi untuk setiap stasiun kerja. Status order yang akan dijadwalkan terdapat pada lampiran.

4.1.6 Waktu proses buku tulis

Waktu proses produksi buku tulis merupakan waktu proses untuk mengerjakan suatu order. Waktu proses buku tulis dari order-order yang akan dijadwalkan terdapat pada lampiran.

4.1.7 Saat siap mesin (*server*) di stasiun kerja bottleneck

Saat siap mesin (*server*) di stasiun kerja *bottleneck* merupakan saat *server* dalam keadaan selesai mengerjakan order dan siap digunakan untuk mengerjakan order lain yang akan dijadwalkan. Saat siap mesin (*server*) ditentukan dari stasiun kerja *bottleneck* karena penjadwalan untuk sistem *bottleneck* diawali dari stasiun kerja *bottleneck*. Saat siap mesin (*server*) di stasiun kerja *bottleneck* (Stasiun Kerja Vurnish Cover) adalah pada tanggal 13/04/2009, jam 15.00, shift ke-2. Saat siap mesin (*server*) di stasiun kerja *bottleneck* ditransformasi menjadi jam ke-*t* yaitu pada jam ke-6.

4.2 Karakterisasi Sistem Penjadwalan Produksi Buku Tulis

PT. Solo Murni merupakan sebuah perusahaan yang bergerak dalam bidang percetakan dan beralamatkan Jl. A.Yani 378 Solo. Produk yang dihasilkan di PT. Solo Murni adalah produk *stationery* yang digolongkan menjadi enam kelompok yaitu, *school supplies*, *office supplies*, *writing papers*, *envelopes*, *gift wrapping set*, dan *others*. Sistem produksi yang digunakan adalah *job order* untuk order jasa dan ekspor, sedangkan untuk produksi lokal dilaksanakan dengan sistem *make to stock*. Akan tetapi secara keseluruhan, berdasarkan kebijakan PT. Solo Murni sistem manufaktur produksi untuk *make to stock* mencapai nilai 95 persen, sedangkan sisanya yaitu 5 persen untuk mencukupi *job order*.

Pada kelompok produk *school supplies*, dalam proses produksinya dibagi menjadi dua yaitu buku tulis tebalan yang melewati jalur mesin Monoblok dan buku tulis jahit kawat yang melewati jalur mesin 321. Penelitian ini difokuskan pada permasalahan penjadwalan produksi buku tulis jahit kawat yang melewati jalur mesin 321 karena produk ini adalah produk unggulan di PT. Solo Murni dan sering terlihat penumpukan di stasiun kerja ketika proses produksi berlangsung.

Buku tulis jahit kawat ini merupakan model varian *repetitive*, yaitu bahwa order tersebut dalam proses produksinya tidak melibatkan unsur perancangan dalam proses manufaktur. Sedangkan karakteristik sistem produksi buku tulis jahit kawat (jalur mesin 321) yaitu *mixed model repetitive*, artinya bahwa lini produksi mengerjakan berbagai jenis model dalam satu lini produksi.

Pada produksi buku tulis pada jalur mesin 321 memiliki tiga bagian produksi, yaitu bagian Pre-Printing, bagian Cetak dan bagian Finishing. Adapun penjelasan bagian dalam proses produksi buku tulis pada jalur mesin 321 tersebut adalah:

1. Bagian Pre-Printing (Pra-Produksi)

Pada tahap ini merupakan tahap sebelum dimulainya proses produksi. Tahap ini terdiri dari proses antara lain:

- a. *Desain*, yaitu menyiapkan desain produk sebagai dasar proses montage. Desain yang dibuat biasanya bercorak lucu, mengikuti *event* tertentu (contohnya *event* piala EURO, piala dunia), kartun, dan lain-lain.
- b. *Montage*, yaitu membuat gambar *montage* sebagai dasar pembuatan plate untuk proses cetak. Pembuatan *film* dan *plate* berdasarkan kalkulasi yang telah dilakukan oleh *scheduler* di department PPIC. Kalkulasi tersebut meliputi bagaimana menentukan jumlah mata yang harus diplotkan dalam satu film sehingga efisien dari segi material maupun biaya. *Layout* mata tersebut juga harus diperhitungkan karena akan berpengaruh pada proses selanjutnya terutama proses lipat.
- c. *Plate*, yaitu proses membuat plate untuk proses cetak. *Plate* tersebut yang nantinya akan dipasang pada mesin cetak. Dalam *plate*, untuk masing-masing warna disiapkan *plate* yang berlainan (satu *plate* satu warna). *Plate* yang disediakan ada dua macam yaitu *plate* untuk *cover* dan isi.

2. Bagian Cetak

Pada bagian ini, *plate* yang sudah dibuat di bagian *montage*, dipasang pada mesin cetak. *Plate* tersebut dipergunakan untuk menyablon kertas untuk isi dan *cover* buku tulis yang ukurannya telah disesuaikan dengan *plate*. Sebelum dicetak, kertas dirapikan dengan mesin potong sisir, lalu proses cetak dilakukan. Untuk hasil cetakan *cover*, setelah jadi masih melalui tahap *varnish* atau proses UV/Glitter. Hal ini bertujuan agar hasil cetakan tersebut awet (tidak luntur atau pudar).

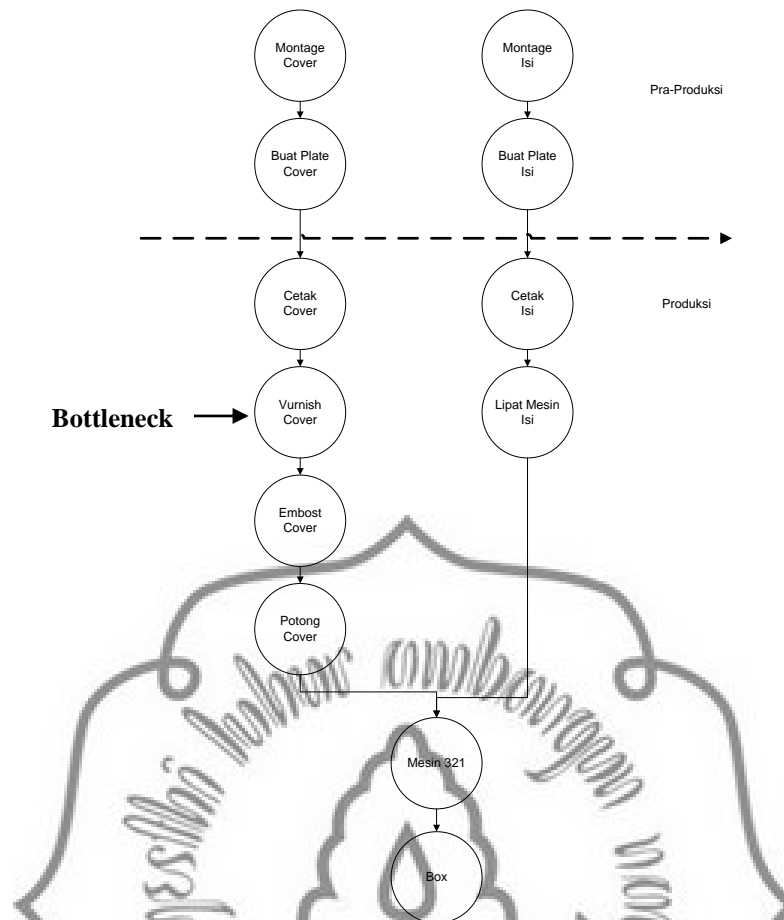
3. Bagian *Finishing*

Setelah proses cetak selesai dilakukan, hasil cetakan dikirim ke bagian *finishing*. Proses tersebut yaitu antara lain:

- a. Lipat, yaitu proses pertama yang dilakukan untuk cetakan isi.
- b. TAB, yaitu proses yang dilakukan khusus untuk produk buku tulis. Proses TAB antara lain: jahit kawat (mesin 321), binding dan flexo.
- c. *Packing*, yaitu proses terakhir dari rangkaian proses-proses yang terjadi.
- d. Bagian pendukung, yang terdiri dari *doos* dan proses *rework*.

Pengerjaan buku tulis pada jalur mesin 321 dilakukan dengan sistem produksi *mixed model repetitive flowshop*. Sistem produksi *flowshop* ditunjukkan dengan pengerjaan order dari stasiun pertama hingga stasiun terakhir dalam proses produksinya dengan tidak mengalami proses balik ke bagian hulu atau bagian produksi sebelumnya. Setiap stasiun kerja pada proses produksi buku tulis pada jalur mesin 321 memiliki jumlah *server* (mesin) yang berbeda-beda.

Uraian proses produksi buku tulis pada jalur mesin 321 diatas dapat disimpulkan seperti pada gambar 4.1 berikut:

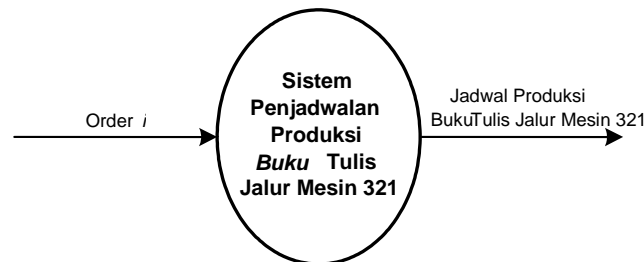


Gambar 4.1 Proses Produksi Buku Tulis 38 Jalur Mesin 321
Sumber: PT.Solo Murni, 2008

Pada penelitian ini yang diteliti hanya pada bagian produksi saja, sedangkan bagian pra-Produksi dan proses *packing* (box) tidak diteliti. Pada bagian pra-Produksi tidak diteliti karena pada bagian tersebut merupakan suatu pekerjaan seni sehingga untuk waktu prosesnya tidak dapat diperhitungkan secara pasti dan untuk proses *packing* (box) di sub-kontrakkan.

Berdasarkan deskripsi umum sistem manufaktur tersebut maka dapat dilakukan karakterisasi sistem penjadwalan produksi buku tulis pada jalur mesin 321. Karakterisasi sistem pada dasarnya merupakan proses simplifikasi atau idealisasi sistem nyata yang sesuai dengan permasalahan yang diteliti (Murthy et. al, 1990). Karakterisasi sistem penjadwalan produksi buku tulis yang melewati jalur mesin 321 dilakukan dengan pendekatan deskripsi *black-box*, yaitu pendeskripsian sistem hanya dilakukan pada variabel-variabel tertentu yang berinteraksi dengan lingkungan sistem sedangkan struktur dalam sistem diabaikan. Karakteristik sistem penjadwalan produksi buku tulis pada jalur mesin

321 pada kelompok produk *office supply* PT. Solo Murni dapat dilihat pada Gambar 4.2.



Gambar 4.2 Karakterisasi Sistem Penjadwalan Produksi Buku tulis pada jalur mesin 321

Berdasarkan karakteristik sistem diketahui bahwa sistem penjadwalan produksi buku tulis pada jalur mesin 321 merupakan sistem terbuka (*open system*). Hal ini ditunjukkan dengan adanya interaksi antara sistem dengan obyek-obyek di lingkungannya. Interaksi ini ditunjukkan dengan adanya *input* dari obyek di luar sistem ke dalam sistem dan adanya *output* dari sistem ke obyek di luar sistem. Obyek yang terlibat dalam sistem penjadwalan produksi buku tulis yang melewati jalur mesin 321 adalah order dan *server* (mesin atau tim kerja).

Pada jalur mesin 321 dalam produksi buku tulis terdapat sistem *bottleneck* yaitu pada Vurnish Cover. Jadi dalam merancang model penjadwalan dengan pendekatan *DBR* ini mempertimbangkan stasiun Vurnish Cover sebagai SKB dan menjadikannya sebagai titik kontrol dalam penjadwalan (Sipper dan Bulfin, 1997). Sesuai pada gambar 4.2 diatas dan mengakomodasi keberadaan SKB (Vurnish Cover), maka mekanisme penjadwalan produksi buku tulis adalah penjadwalan yang dilakukan pertama kali adalah pada stasiun Vurnish Cover. Selanjutnya untuk Cetak Cover yang merupakan stasiun kerja sebelum SKB (Vurnish Cover), penjadwalannya dilakukan dengan *backward scheduling* yaitu dengan menarik dari jadwal SKB (Vurnish Cover). Dan untuk stasiun kerja Embost Cover, Potong Cover dan Mesin 321 dalam penjadwalannya dilakukan dengan *forward scheduling* yaitu dengan mendorong maju dari jadwal SKB (Vurnish Cover). Sedangkan pada stasiun kerja Cetak Isi dan Lipat Isi dalam penjadwalannya dilakukan dengan dengan *backward scheduling*. Akan tetapi terdapat perbedaan antara penjadwalan di stasiun Cetak Cover dan stasiun Cetak

Isi dan Lipat Isi yaitu dari mana penarikan jadwal yang dilakukan. Pada stasiun Cetak Isi dan Lipat Isi, jadwal ditarik dari hasil jadwal di stasiun mesin 321.

Rancangan model penjadwalan produksi buku tulis pada jalur mesin 321 menggunakan pendekatan *DBR* dengan kriteria minimasi *mean flowtime*, terdapat variabel-variabel maupun parameter-parameter yang berperan. Variabel adalah atribut yang diperlukan untuk mendeskripsikan interaksi antar obyek (Murthy et. al, 1990). Variabel-variabel yang berperan dalam sistem penjadwalan produksi buku tulis pada jalur mesin 321 adalah:

1. Saat siap *server s* di stasiun kerja j (R_{sj})

Saat siap *server s* (mesin atau tim kerja) di stasiun kerja j merupakan variabel yang mendeskripsikan waktu *server s* di stasiun kerja j siap memproses order.

2. *Mean lead time* order i di stasiun kerja j (ET_{ij})

Mean lead time order i di stasiun kerja j merupakan variabel yang mendeskripsikan lamanya *lead time* order i di stasiun kerja j . *Mean lead time* order i di stasiun kerja j dapat ditentukan dengan Algoritma Zijm-Buitenhok.

3. *Earliest bottleneck release* order i ($r_i^{(b)}$)

Earliest bottleneck release order i merupakan variabel yang mendeskripsikan saat tercepat order i bisa mulai diproses di SKB. *Earliest bottleneck release* order i bisa ditentukan dengan menjumlahkan *mean lead time* order i di stasiun kerja j sebelum SKB.

4. *Bottleneck due date* order i ($d_i^{(b)}$)

Bottleneck due date order i merupakan variabel yang mendeskripsikan saat paling akhir order i harus selesai diproses SKB. *Bottleneck due date* order i bisa ditentukan dengan mengurangi d_i dengan jumlah *mean lead time* order i di stasiun kerja j setelah SKB.

5. *Bottleneck slack time* ($ST_i^{(b)}$)

Bottleneck slack time order i merupakan variabel yang mendeskripsikan saat *slack time* order i pada SKB. *Bottleneck slack time* order i bisa ditentukan dengan mengurangi d_i dengan *present time* dan *processing time* di SKB.

6. *Slack time* order i di stasiun kerja j (ST_{ij})

Slack time order i di stasiun kerja j merupakan variabel yang mendeskripsikan saat *slack time* order i di stasiun kerja j . *Slack time* order i di stasiun kerja j bisa ditentukan dengan mengurangi d_i dengan *completion time* order di stasiun kerja $(j - 1)$ dan *processing time* di stasiun kerja j .

7. Saat *release* order i di stasiun kerja j (r_{iksj})

Saat *release* order i di stasiun kerja j merupakan variabel yang mendeskripsikan saat *release* order i di setiap stasiun kerja j .

8. *Completion time* order i urutan ke- k pada *server* s di stasiun kerja j (c_{iksj})

Completion time order i urutan ke- k pada *server* s di stasiun kerja j merupakan variabel yang mendeskripsikan saat order i selesai dikerjakan di stasiun kerja j .

Parameter adalah atribut intrinsik obyek (Murthy et. al, 1990). Parameter-parameter yang berperan dalam sistem penjadwalan produksi buku tulis pada jalur mesin 321 adalah:

1. Jumlah order i (n)

Jumlah order i merupakan parameter yang mendeskripsikan banyaknya order yang akan diproses di rantai produksi. Jumlah order yang dijadwalkan bersifat deterministik untuk setiap periode penjadwalan karena jumlah order diketahui pada awal periode penjadwalan produksi (t_0). Divisi PPIC menjadwalkan suatu order hanya jika kelengkapan dan kebutuhan yang mendukung pengerjaan order di rantai produksi sudah siap.

2. Jumlah stasiun kerja di lini produksi (N)

Jumlah stasiun kerja di lini produksi merupakan parameter yang menunjukkan banyaknya stasiun kerja yang dilalui order i hingga selesai diproses. Parameter jumlah stasiun kerja di lini produksi merupakan parameter yang bersifat deterministik karena jumlah *server* bisa dihitung secara pasti.

3. Jumlah *server* di stasiun kerja j (m_j)

Jumlah *server* di stasiun kerja j merupakan parameter yang menentukan kapasitas produksi. Kapasitas produksi akan menentukan penjadwalan

produksi. Parameter jumlah *server* di stasiun kerja j merupakan parameter yang bersifat deterministik karena jumlah *server* bisa dihitung secara pasti.

4. Saat masuk order i (I_i)

Saat masuk order i merupakan parameter yang mendeskripsikan waktu kedatangan order i untuk siap diproses di rantai produksi. Saat masuk order i merupakan parameter yang bersifat deterministik karena saat masuk order i bisa ditentukan 3-7 hari setelah tanggal pemesanan dilakukan (masa persiapan produksi).

5. *Due date* order i (d_i)

Due date order i merupakan parameter yang menentukan waktu order i harus selesai dikerjakan. Parameter ini merupakan parameter yang bersifat deterministik karena ditentukan berdasarkan kesepakatan dengan konsumen.

6. Waktu proses order i di stasiun kerja j (p_{ij})

Waktu proses order i di stasiun kerja j merupakan parameter yang menunjukkan lama waktu proses order. Parameter waktu proses order i di stasiun kerja j merupakan parameter yang bersifat deterministik karena waktu yang digunakan untuk memproduksi bisa dihitung secara pasti.

7. Periode awal order i dijadwalkan di lini produksi (t_0)

Parameter periode awal order i dijadwalkan di lini produksi merupakan parameter yang bersifat deterministik karena periode awal order i dijadwalkan di lini produksi ditentukan setiap minggu sekali yaitu pada awal minggu.

8. Saat siap *server* s di stasiun kerja *bottleneck* ($R_{sj(j=b)}$) (awal)

Parameter ini mendeskripsikan saat *server* s di stasiun kerja *bottleneck* sudah menyelesaikan order pada periode penjadwalan sebelumnya. Parameter saat siap *server* s di stasiun kerja *bottleneck* merupakan parameter yang bersifat deterministik karena bisa diketahui dari penjadwalan periode sebelumnya.

9. Ukuran *lot* order i (Q_i)

Ukuran *lot* order i merupakan parameter yang mendeskripsikan banyaknya order yang dikerjakan dalam satu *lot*. Parameter ukuran *lot* order i bisa ditentukan dengan pasti, yaitu 1 ($Q_i = 1$). Dengan demikian parameter ini bersifat deterministik.

4.3 Pengembangan Algoritma Penjadwalan Produksi Buku Tulis Dengan Pendekatan *Drum-Buffer-Rope* (DBR)

Algoritma penjadwalan *DBR* dikembangkan dari model *bottleneck scheduling* yang telah dilakukan Sipper dan Bulfin (1997). Model dasar yang dikembangkan Sipper dan Bulfin (1997) mengasumsikan tidak ada *waiting time* order di setiap stasiun kerja sehingga *lead time* order sebelum SKB merupakan jumlah waktu proses order di setiap stasiun kerja sebelum SKB, sedangkan *lead time* order sesudah SKB merupakan jumlah waktu proses order di setiap stasiun kerja setelah SKB. Pada penelitian ini perhitungan *lead time* order sebelum dan sesudah SKB dilakukan dengan Algoritma Zijm-Buitenhok. Algoritma Zijm-Buitenhok merupakan algoritma untuk memperkirakan *mean lead time* yang mencakup *waiting time* dan waktu proses order berdasarkan teori antrian. Minimasi *mean flowtime* dilakukan dengan pendekatan SPT (*Short Processing Time*).

Pengembangan penjadwalan produksi buku tulis dengan pendekatan *DBR* ini dilakukan dengan beberapa penyesuaian agar bisa digunakan untuk memecahkan permasalahan nyata di perusahaan. Penjadwalan produksi buku tulis dengan pendekatan *DBR* dengan kriteria minimasi *mean flowtime* dikembangkan dengan tahap-tahap sebagai berikut:

1. Perhitungan *mean lead time* order i di stasiun kerja j (ET_{ij}).
2. Penentuan *earliest bottleneck release* order i ($r_i^{(b)}$) dan *bottleneck due date* order i ($d_i^{(b)}$).
3. Pendistribusian order i ke *server-server* di stasiun kerja *bottleneck* (SKB).
4. Pemeriksaan kelayakan jadwal inisial *server* s di SKB.
5. Revisi jadwal inisial *server* s tidak layak di SKB dengan SPT (*Shortest Processing Time*) untuk meminimasi *mean flowtime*.
6. Penjadwalan produksi di stasiun kerja j .
 - a) Penjadwalan produksi di stasiun kerja *bottleneck*.
 - b) Penjadwalan produksi di stasiun kerja *upstream* (*Backward Scheduling*).
 - c) Penjadwalan produksi di stasiun kerja *downstream* (*Forward Scheduling*).

7. Pemeriksaan ketersediaan order i' (order sisipan) dan atau order i periode berikutnya ($t = t_0 + 1$)

8. Pengukuran performansi.

Notasi:

| | |
|----------------|---|
| n | : Jumlah order i |
| i | : Order i ($i = 1, 2, \dots, n$) |
| N | : Jumlah stasiun kerja di lini produksi |
| j | : Stasiun kerja j ($j = 1, 2, \dots, N$) |
| b | : Stasiun kerja <i>bottleneck</i> (SKB) |
| m_j | : Jumlah <i>server</i> di stasiun kerja j |
| s | : <i>Server</i> s ($s = 1, 2, \dots, m_j$) |
| $r_i^{(b)}$ | : <i>Earliest bottleneck release</i> order i |
| $d_i^{(b)}$ | : <i>Bottleneck due date</i> order i |
| I_i | : Saat masuk order i pada Divisi <i>School Supplies</i> |
| d_i | : <i>Due date</i> order i |
| t_0 | : Periode awal order i dijadwalkan di lini produksi |
| p_{ij} | : Waktu proses order i di stasiun kerja j (jam) |
| τ_{ij} | : Waktu setup operasi k order i di stasiun kerja j |
| a_{ij} | : Waktu proses per unit order i di stasiun kerja j |
| Q_i | : Ukuran <i>lot</i> order i |
| D_i | : <i>Demand rate</i> order I (unit/jam) |
| δ_{ij} | : 1, jika order i dikerjakan di stasiun kerja j 0, jika yang lainnya |
| λ_{ij} | : <i>Arrival rate</i> order i stasiun kerja j (unit/jam) |
| λ_j | : <i>Arrival rate</i> stasiun kerja j (unit/jam) |
| ρ_j | : <i>Average work load</i> di stasiun kerja j (unit) |
| W_j | : <i>Mean waiting time</i> di stasiun kerja j (jam) |
| ET_{ij} | : <i>Mean lead time</i> order i di stasiun kerja j (jam) |

- q : Jumlah order i pada server s di stasiun kerja j
 k : Order i urutan ke- k pada server s di stasiun kerja j ($k = 1, 2, \dots, q$)
 R_{sj} : Saat siap server s di stasiun kerja j
 $ST_i^{(b)}$: *Bottleneck slack time* order i
 ST_{ij} : *Slack time* order i di stasiun kerja j
 r_{iksj} : Saat *release* order i urutan ke- k pada server s di stasiun kerja j (jam ke-)
 c_{iksj} : *Completion time* order i urutan ke- k pada server s di stasiun kerja j (jam ke-)
 SL : *Scheduled list*
 UL : *Unscheduled list*
 F_i : *Flowtime* sistem keseluruhan (jam)
 \bar{F} : *Mean Flowtime* sistem keseluruhan (jam)

4.3.1 Perhitungan *mean lead time* order i di stasiun kerja j (ET_{ij})

Adapun tahap pertama yaitu perhitungan *mean lead time* order i di stasiun kerja j (ET_{ij}). ET_{ij} dalam produksi buku tulis pada jalur mesin 321 dilakukan dengan langkah-langkah sebagai berikut:

Perhitungan *Demand Rate* Order i (D_i)

Adapun langkah-langkah untuk menentukan *demand rate* order i (D_i) adalah sebagai berikut:

Langkah 0 : PPIC bagian TAB mengambil data order i (buku tulis pada jalur mesin 321) yang akan dijadwalkan (order i dengan $I_i < t_0$), *due date* order i (d_i), periode awal order i dijadwalkan di rantai produksi (t_0), dan jam kerja regular.

Langkah 1 : Tetapkan $i = 1$.

Langkah 2 : Hitung *demand rate* order i dengan persamaan (4.1):

$$D_i = \frac{1}{PLT} = \frac{1}{d_i - t_0} \quad (\text{unit/jam})$$

Langkah 3 : Jika $i = n$, lanjutkan ke langkah 4. Lainnya hitung $i = i + 1$ dan

kembali ke langkah 2.

Langkah 4 : Simpan *demand rate* order i (D_i).

Perhitungan *Arrival Rate* Order i (λ_{ij})

Adapun langkah-langkah untuk menentukan *arrival rate* order i di stasiun kerja j (λ_{ij}) adalah sebagai berikut:

Langkah 0 : PPIC bagian TAB mengambil data order i yang akan dijadwalkan, *demand rate* order i (D_i), status order i dikerjakan di stasiun kerja j (δ_{ij}), dan jumlah *server* di stasiun kerja j (m_j).

Langkah 1 : Tetapkan $i = 1, j = 1$.

Langkah 2 : Hitung *arrival rate* order i di stasiun kerja j dengan persamaan (4.2): $\lambda_{ij} = \frac{D_i}{m_j} \cdot \delta_{ij}$ (unit/jam)

Langkah 3 : Jika $i = n$, lanjutkan ke langkah 4. Lainnya hitung $i = i + 1$ dan kembali ke langkah 2.

Langkah 4 : Jika $j = N$, lanjutkan ke langkah 5. Lainnya hitung $j = j + 1$ dan kembali ke langkah 2.

Langkah 5 : Simpan *arrival rate* order i di stasiun kerja j (λ_{ij}).

Perhitungan *Average Workload* di Stasiun Kerja j (ρ_j)

Adapun langkah-langkah untuk menentukan *Average Workload* di Stasiun Kerja j (ρ_j) adalah sebagai berikut:

Langkah 0 : PPIC bagian TAB mengambil data order i yang akan dijadwalkan, *arrival rate* order i di stasiun kerja j (λ_{ij}), dan waktu proses order i di stasiun kerja j (p_{ij}).

Langkah 1 : Tetapkan $j = 1$.

Langkah 2 : Hitung *average work load* di stasiun kerja j dengan persamaan (4.3): $\rho_j = \sum_{i=1}^n \lambda_{ij} \cdot p_{ij}$ (unit)

Sedemikian rupa sehingga $\rho_j < 1$.

Jika $\rho_j \geq 1$ tetapi masih ada order i yang belum dihitung, keluarkan order i dari kelompok order i yang akan dijadwalkan. Simpan order i tersebut untuk periode penjadwalan berikutnya.

Langkah 3 : Jika $j = N$, lanjutkan ke langkah 4. Lainnya hitung $j = j + 1$ dan kembali ke langkah 2.

Langkah 4 : Simpan *average workload* di stasiun kerja j (ρ_j).

Perhitungan *Mean Waiting Time* di Stasiun Kerja j (W_j)

Adapun langkah-langkah untuk menentukan *Mean Waiting Time* di Stasiun Kerja j (W_j) adalah sebagai berikut:

Langkah 0 : PPIC bagian TAB mengambil data order i yang akan dijadwalkan, *arrival rate* order i di stasiun kerja j (λ_{ij}), waktu proses order i di stasiun kerja j (p_{ij}), dan *average workload* di stasiun kerja j (ρ_j).

Langkah 1 : Tetapkan $j = 1$.

Langkah 2 : Hitung *mean waiting time* di stasiun kerja j dengan persamaan (4.4):

$$EW_j = \frac{\sum_{i=1}^n \lambda_{ij} \cdot (p_{ij})^2}{2(1 - \rho_j)} \quad (\text{jam})$$

Langkah 3 : Jika $j = N$, lanjutkan ke langkah 4. Lainnya hitung $j = j + 1$ dan kembali ke langkah 2.

Langkah 4 : Simpan *mean waiting time* di stasiun kerja j (EW_j).

Perhitungan *Mean Lead Time* Order i di Stasiun Kerja j (ET_{ij})

Adapun langkah-langkah untuk menentukan *mean lead time* order i di stasiun kerja j (ET_{ij}) adalah sebagai berikut:

Langkah 0 : PPIC bagian TAB mengambil data order i yang akan dijadwalkan, *mean waiting time* di stasiun kerja j (EW_j), status order i dikerjakan di stasiun kerja j (δ_{ij}), dan waktu proses order

i di stasiun kerja j (p_{ij}).

Langkah 1 : Tetapkan $i = 1, j = 1$.

Langkah 2 : Hitung *mean lead time* order i di stasiun kerja j dengan persamaan (4.5):

$$ET_{ij} = EW_j \cdot \delta_{ij} + p_{ij} \quad (\text{jam})$$

Langkah 3 : Jika $i = n$, lanjutkan ke langkah 4. Lainnya hitung $i = i + 1$ dan kembali ke langkah 2.

Langkah 4 : Jika $j = N$, lanjutkan ke langkah 5. Lainnya hitung $j = j + 1$ dan kembali ke langkah 2.

Langkah 5 : Simpan *mean lead time* order i di stasiun kerja j (ET_{ij}).

4.3.2 Penentuan *earliest bottleneck release order i* ($r_i^{(b)}$) dan *bottleneck due date order i* ($d_i^{(b)}$)

Pada penjadwalan produksi buku tulis dengan pendekatan *DBR* dengan kriteria minimasi *mean flowtime* ini, setelah diketahui *mean lead time* produksi buku tulis pada jalur mesin 321 maka selanjutnya dilakukan tahap penentuan *earliest bottleneck release order i* ($r_i^{(b)}$) dan *bottleneck due date order i* ($d_i^{(b)}$).

Adapun langkah-langkah untuk menentukan *Earliest bottleneck release order i* ($r_i^{(b)}$) dan *Bottleneck due date order i* ($d_i^{(b)}$) adalah sebagai berikut:

Langkah 0 : PPIC bagian TAB mengambil data order i yang akan dijadwalkan order i yang akan dijadwalkan (order i dengan $I_i < t_0$), *due date* order i (d_i), *mean lead time* order i di stasiun kerja j (ET_{ij}), dan periode awal order i dijadwalkan di rantai produksi (t_0).

Langkah 1 : Tetapkan $i = 1, j = b$.

Langkah 2 : Hitung $r_i^{(b)}$ (*earliest bottleneck release order i*) dan $d_i^{(b)}$ (*bottleneck due date order i*).

$$r_i^{(b)} = t_0 + \sum_{j=1}^{b-1} ET_{ij} \quad (\text{jam})$$

$$d_i^{(b)} = d_i - \sum_{j=b+1}^N ET_{ij} \quad (\text{jam})$$

Langkah 3 : Jika $i = n$, lanjutkan ke langkah 5. Lainnya hitung $i = i + 1$ dan kembali ke langkah 3.

Langkah 4 : Simpan data *earliest bottleneck release* order i ($r_i^{(b)}$) dan *bottleneck due date* order i ($d_i^{(b)}$).

4.3.3 Pendistribusian order i ke *server-server* di stasiun kerja *bottleneck* (SKB)

Setelah *earliest bottleneck release* order i ($r_i^{(b)}$) dan *bottleneck due date* order i ($d_i^{(b)}$) ditentukan tahapan selanjutnya adalah mendistribusikan order-order i ke *server-server* di SKB.

Adapun langkah-langkah untuk pendistribusian order i ke *server-server* di SKB adalah sebagai berikut:

Langkah 0 : PPIC bagian TAB mengambil data order i yang akan dijadwalkan, *earliest bottleneck release* order i ($r_i^{(b)}$), *bottleneck due date* order i ($d_i^{(b)}$), waktu proses order i di stasiun kerja j (p_{ij}), dan saat siap *server* s di stasiun kerja j (R_{sj}).

Langkah 1 : Tetapkan $j = b$.

Langkah 2 : Pilih *server* s dengan R_{sj} minimum pada stasiun kerja j . Jika R_{sj} minimum > 1 , pilih R_{sj} minimum pada *server* s dengan nomor urut terkecil ($s = 1, 2, \dots, m_j$).

Langkah 3 : Pilih order i dengan $r_i^{(b)} \leq R_{sj}$ dari himpunan data order i yang akan dijadwalkan dan memiliki *bottleneck slack time* ($ST_i^{(b)}$) tercepat yang meminimasi rata-rata *flowtime* penyelesaian (*mean flowtime*) order i di stasiun kerja j .

$$ST_i^{(b)} = d_i^{(b)} - R_{sj} - p_{ij} \quad (\text{jam ke-})$$

Jika $ST_i^{(b)}$ tercepat > 1 , pilih order i dengan $d_i^{(b)}$ tercepat. Jika $d_i^{(b)}$ tercepat > 1 , pilih order i berdasarkan p_{ij} terpendek. Jika p_{ij} terpendek > 1 , pilih order i berdasarkan nomor urut order i

terkecil. Jika tidak ada order dengan kriteria $r_i^{(b)} \leq R_{sj}$ maka pilih order i dengan $r_i^{(b)} - R_{sj}$ minimum.

Tempatkan order i terpilih pada *server* s terpilih.

Langkah 4 : Hitung kembali saat siap *server* s di stasiun kerja j setelah penempatan order i terpilih di *server* s terpilih.

$$R_{sj} = R_{sj} + p_{ij} \quad (\text{jam ke-})$$

Langkah 5 : Jika himpunan order i yang akan dijadwalkan = \emptyset , lanjutkan ke langkah 6. Lainnya kembali ke langkah 2.

Langkah 6 : Kelompokkan order-order tersebut berdasarkan *server*-nya kemudian simpan order i yang telah terjadwal pada *server* s sebagai jadwal inisial *server* s .

4.3.4 Pemeriksaan kelayakan jadwal inisial *server* s di SKB

Tahap selanjutnya setelah order-order i didistribusikan yaitu memeriksa kelayakan pendistribusian order i (jadwal inisial) di SKB. Kelayakan pendistribusian order i ini dinyatakan layak jika *completion time* order i kurang atau sama dengan *due date*-nya.

Adapun langkah-langkah untuk memeriksa kelayakan pendistribusian order i (jadwal inisial) di SKB adalah sebagai berikut:

Langkah 0 : PPIC bagian TAB mengambil data *bottleneck due date* order i ($d_i^{(b)}$), waktu proses order i di stasiun kerja j (p_{ij}), saat siap *server* s di stasiun kerja j (R_{sj}), dan jadwal inisial *server* s .

Langkah 1 : Tetapkan $s = 1, j = b$.

Langkah 2 : Tetapkan $k = 1$.

Langkah 3 : Hitung *completion time* order i urutan ke- k pada *server* s di stasiun kerja j .

$$c_{iksj} = R_{sj} + p_{ij} \quad (\text{jam ke-})$$

Langkah 4 : Tetapkan $k = 2$.

Langkah 5 : Hitung $c_{iksj} = c_{i(k-1)sj} + p_{ij}$ (jam ke-)

Langkah 6 : Jika $k = q$, lanjutkan ke langkah 7. Lainnya hitung $k = k + 1$ dan

kembali ke langkah 5.

Langkah 7 : Tulis kembali jadwal inisial *server s*, p_{ij} , $d_i^{(b)}$, dan c_{iksj} .

Langkah 8 : Periksa kelayakan jadwal inisial *server s*. Untuk semua order i pada *server s*, jika $c_{iksj} \leq d_i^{(b)}$ maka jadwal inisial *server s* layak. Lainnya jadwal inisial *server s* tidak layak.

Langkah 9 : Jika $s = m_j$, lanjutkan ke langkah 10. Lainnya hitung $s = s + 1$ dan kembali ke langkah 2.

Langkah 10 : Simpan jadwal inisial *server s* layak sebagai jadwal produksi *server s* di stasiun kerja j dan masukkan jadwal inisial *server s* tidak layak pada himpunan jadwal inisial *server s* tidak layak.

4.3.5 Revisi jadwal inisial *server s* tidak layak di SKB dengan SPT (*Shortest Processing Time*) untuk meminimasi *mean flowtime*

Jika setelah dilakukan pemeriksaan kelayakan jadwal inisial terdapat order yang masih tidak layak, maka jadwal inisial tersebut dilakukan revisi atau perbaikan dengan menggunakan metode SPT (*Shortest Processing Time*) untuk optimasi dalam meminimasi *mean flowtime*.

Adapun langkah-langkah untuk Revisi Jadwal Inisial *Server s* Tidak Layak di SKB adalah sebagai berikut:

Langkah 0 : PPIC bagian TAB mengambil data jadwal inisial *server s*, himpunan jadwal inisial *server s* tidak layak, *bottleneck due date* order i dan waktu proses order i di stasiun kerja j (p_{ij}).

Langkah 1 : Tetapkan $s = 1, j = b$.

Langkah 2 : Panggil kembali jadwal inisial *server s*. Jika jadwal inisial *server s* \in himpunan jadwal inisial *server s* tidak layak, lanjutkan ke langkah 3. Lainnya lanjutkan ke langkah 7.

Langkah 3 : Tulis kembali jadwal inisial *server s*, p_{ij} , dan $d_i^{(b)}$.

Langkah 4 : Tetapkan jadwal inisial *server s* = UL ,

Langkah 5 : Tetapkan pasangan order i (urutan ke-1 dan ke-2) dari jadwal inisial *server s*.

Langkah 6 : Tukar order i dalam urutan dimana p_{ij} order terkecil dalam urutan

yang pertama (tetapkan urutan pertama untuk order i dengan p_{ij} terpendek).

Jika order pada $UL = \emptyset$, tempatkan order i pada SL dan lanjutkan ke langkah 7. Lainnya ulang ke langkah 5 hingga urutan pengerjaan order terbentuk..

Langkah 7 : Jika $s = m_j$, lanjutkan ke langkah 8. Lainnya hitung $s = s + 1$ dan kembali ke langkah 2.

Langkah 8 : Simpan jadwal insial *server* s yang sudah direvisi sebagai jadwal insial revisi *server* s di stasiun kerja j .

4.3.6 Penjadwalan produksi di stasiun kerja j

Selanjutnya dilakukan penjadwalan produksi buku tulis dengan mempertimbangkan SKB sebagai stasiun konstrain. Pada penjadwalan produksi buku tulis di stasiun kerja j meliputi tiga bagian, yaitu

- 1) Penjadwalan produksi di stasiun kerja *bottleneck*.
- 2) Penjadwalan produksi di stasiun kerja *upstream* (*Backward Scheduling*).
- 3) Penjadwalan produksi di stasiun kerja *downstream* (*Forward Scheduling*).

Algoritma a.1 : Penjadwalan Produksi di SKB (Stasiun Kerja Vurnish Cover)

Langkah 0 : PPIC bagian TAB mengambil data jadwal insial *server* s layak, jadwal insial revisi *server* s di stasiun kerja j , saat siap *server* s di stasiun kerja j (R_{sj}), dan waktu proses order i di stasiun kerja j (p_{ij}).

Langkah 1 : Tetapkan $j = b$, $s = 1$.

Langkah 2 : Tetapkan $k = 1$.

Langkah 3 : Hitung saat *release* dan *completion time* order i urutan ke- k pada *server* s di stasiun kerja j berdasarkan jadwal insial *server* s layak dan jadwal insial revisi *server* s di stasiun kerja j .

$$r_{iksj} = R_{sj} \quad (\text{jam ke-})$$

$$c_{iksj} = r_{iksj} + p_{ij} \quad (\text{jam ke-})$$

Langkah 4 : Tetapkan $k = 2$.

Langkah 5 : Lanjutkan kembali perhitungan saat *release* dan *completion time* order i urutan ke- k pada *server* s di stasiun kerja j .

$$r_{iks_j} = c_{i(k-1)s_j} \quad (\text{jam ke-})$$

$$c_{iks_j} = r_{iks_j} + p_{ij} \quad (\text{jam ke-})$$

Langkah 6 : Jika $k = q$, lanjutkan ke langkah 7. Lainnya hitung $k = k + 1$ dan kembali ke langkah 5.

Langkah 7 : Jika $s = m_j$, lanjutkan ke langkah 8. Lainnya hitung $s = s + 1$ dan kembali ke langkah 2.

Langkah 8 : Simpan saat *release* order i pada *server* s di stasiun kerja j sebagai jadwal produksi di stasiun kerja j dan dalam himpunan saat *release* order i di stasiun kerja j . Simpan *completion time* order i urutan ke- k pada *server* s di stasiun kerja j sebagai himpunan *completion time* order i di stasiun kerja j .

Algoritma a.2 : Penjadwalan Produksi di SKNB Upstream (Backward Scheduling)

Langkah 0 : PPIC bagian TAB mengambil data himpunan saat *release* order i di stasiun kerja j dan waktu proses order I di stasiun kerja j (p_{ij}).

Langkah 1 : Tetapkan $j = b - 1$.

Langkah 2 : Pilih order i dengan $r_{iks(j+1)}$ terlama dalam himpunan saat *release* order i di stasiun kerja $(j + 1)$. Jika $r_{iks(j+1)}$ terlama > 1 , pilih order i dengan p_{ij} terpanjang. Jika p_{ij} terpanjang > 1 , pilih order i berdasarkan nomor urut order i terkecil.

Langkah 3 : Tetapkan $r_{iks_j}(\text{order } i \text{ yang terakhir masuk di server } s) = r_{iks(j+1)}(\text{order } i \text{ terpilih})$ untuk *server* s di stasiun kerja j yang masih kosong (tidak sedang mengerjakan order) dan panggil kembali r_{iks_j} dari setiap order i yang telah ditempatkan pada *server* s di stasiun kerja j .

Langkah 4 : Tetapkan $s = 1$.

Langkah 5 : Bandingkan $r_{iks(j+1)}(\text{order } i \text{ terpilih})$ dan $r_{iks_j}(\text{order } i \text{ yang terakhir masuk di server } s)$.

Jika $r_{iks(j+1)(order\ i\ terpilih)} \leq r_{iks(j)(order\ i\ yang\ terakhir\ masuk\ di\ server\ s)}$, pilih *server* s sebagai *server* s terpilih dan lanjutkan ke langkah 8. Lainnya hitung selisih $r_{iks(j)(order\ i\ yang\ terakhir\ masuk\ di\ server\ s)}$ dan $r_{iks(j+1)(order\ i\ terpilih)}$:

$$\Delta W_s = r_{iks(j+1)(order\ i\ terpilih)} - r_{iks(j)(order\ i\ yang\ terakhir\ masuk\ di\ server\ s)} \quad (3.20)$$

Lanjutkan ke langkah 6.

Langkah 6 : Jika $s = m_j$, lanjutkan ke langkah 7. Lainnya hitung $s = s + 1$ dan kembali ke langkah 5.

Langkah 7 : Pilih *server* s dengan ΔW_s paling minimum sebagai *server* s terpilih. Jika *server* s dengan ΔW_s paling minimum > 1 , tetapkan *server* s berdasarkan nomor urut terkecil ($s = 1, 2, \dots, m_j$). Lanjutkan ke langkah 9.

Langkah 8 : Tempatkan order i terpilih pada *server* s terpilih dan tetapkan $C_{iks(j)(order\ i\ terpilih)} = r_{iks(j+1)(order\ i\ terpilih)}$. Lanjutkan ke langkah 10.

Langkah 9 : Tempatkan order i terpilih pada *server* s terpilih dan tetapkan $C_{iks(j)(order\ i\ terpilih)} = r_{iks(j)(order\ i\ yang\ terakhir\ masuk\ di\ server\ s)}$. Lanjutkan ke langkah 10.

Langkah 10 : Untuk *server* s terpilih hitung kembali

$$r_{iks(j)} = C_{iks(j)(order\ i\ terpilih)} - p_{ij} \quad (\text{jam ke-})$$

Langkah 11 : Keluarkan order i terpilih dari himpunan saat *release* order i di stasiun kerja $(j + 1)$. Jika himpunan saat *release* order i di stasiun kerja $(j + 1) = \emptyset$, lanjutkan ke langkah 12. Lainnya kembali ke langkah 2.

Langkah 12 : Jika $j = 0$, lanjutkan ke langkah 13. Lainnya hitung $j = j - 1$ dan kembali ke langkah 2.

Langkah 13 : Kelompokkan order-order tersebut berdasarkan *server*-nya, kemudian simpan saat *release* order i pada *server* s di stasiun kerja j (stasiun kerja *upstream*) sebagai jadwal produksi di stasiun kerja j .

Algoritma a.3 : Penjadwalan Produksi di SKNB Downstream (Forward Scheduling)

Langkah 0 : PPIC bagian TAB mengambil data *due date* order i (d_i), waktu proses order i di stasiun kerja j (p_{ij}), himpunan *completion time* order i di stasiun kerja j .

Langkah 1 : Tetapkan $j = b + 1$.

Langkah 2 : Pilih order i dengan $c_{iks(j-1)}$ tercepat dalam himpunan *completion time* order i di stasiun kerja $(j - 1)$. Jika $c_{iks(j-1)}$ tercepat > 1 , pilih order i dengan ST_{ij} tercepat.

$$ST_{ij} = d_i - c_{iks(j-1)} - p_{ij} \quad (\text{jam ke-})$$

Jika ST_{ij} tercepat > 1 , pilih order i dengan d_i tercepat. Jika d_i tercepat > 1 , pilih order i berdasarkan p_{ij} terpendek. Jika p_{ij} terpendek > 1 , pilih order i berdasarkan nomor urut order i terkecil.

Langkah 3 : Tetapkan $R_{sj} = c_{iks(j-1)(\text{order } i \text{ terpilih})}$ untuk *server* s di stasiun kerja j yang masih kosong (tidak sedang mengerjakan order) dan panggil kembali R_{sj} untuk setiap *server* s di stasiun kerja j yang sudah terisi order i .

Langkah 4 : Tetapkan $s = 1$.

Langkah 5 : Bandingkan $c_{iks(j-1)(\text{order } i \text{ terpilih})}$ dengan R_{sj} .

Jika $c_{iks(j-1)(\text{order } i \text{ terpilih})} \geq R_{sj}$, pilih *server* s sebagai *server* s terpilih dan lanjutkan ke langkah 8. Lainnya hitung selisih $c_{iks(j-1)(\text{order } i \text{ terpilih})}$ order i terpilih dengan R_{sj} :

$$\Delta T_s = R_{sj} - c_{iks(j-1)(\text{order } i \text{ terpilih})} \quad (\text{jam ke-})$$

Lanjutkan ke langkah 6.

Langkah 6 : Jika $s = m_j$, lanjutkan ke langkah 7. Lainnya hitung $s = s + 1$ dan kembali ke langkah 5.

Langkah 7 : Pilih *server* s dengan ΔT_s paling minimum sebagai *server* s

terpilih. Jika *server s* dengan ΔT_s paling minimum > 1 , tetapkan *server s* berdasarkan nomor urut terkecil ($s = 1, 2, \dots, m_j$).

Lanjutkan ke langkah 9.

Langkah 8 : Tempatkan order *i* terpilih pada *server s* terpilih dan tetapkan

$$r_{iksj} = c_{iks(j-1)(order\ i\ terpilih)} \cdot \text{Lanjutkan ke langkah 10.}$$

Langkah 9 : Tempatkan order *i* terpilih pada *server s* terpilih dan tetapkan

$$r_{iksj} = c_{iksj(order\ i\ yang\ terakhir\ masuk\ di\ server\ s)} \cdot \text{Lanjutkan ke langkah 10.}$$

Langkah 10 : Hitung kembali:

$$c_{iksj} = r_{iksj} + p_{ij} \quad (\text{jam ke-})$$

Untuk *server s* terpilih, tetapkan kembali $R_{sj} = c_{iksj}$.

Langkah 11 : Keluarkan order *i* terpilih dari himpunan *completion time* order *i* di stasiun kerja ($j - 1$). Jika himpunan *completion time* order *i* di stasiun kerja ($j - 1$) = \emptyset , lanjutkan ke langkah 12. Lainnya kembali ke langkah 2.

Langkah 12 : Kelompokkan order-order tersebut berdasarkan *server*-nya, kemudian simpan saat *release* order *i* pada *server s* di stasiun kerja *j* (stasiun kerja *downstream*) sebagai jadwal produksi di stasiun kerja *j*.

Langkah 13 : Jika $j = N$, berhenti. Lainnya hitung $j = j + 1$ dan kembali ke langkah 2.

4.3.7 Pemeriksaan keterdapatan order *i*' (order sisipan) dan atau order *i* periode berikutnya ($t = t_0 + 1$)

Jika algoritma penjadwalan order *i* untuk periode minggu I diatas telah selesai, selanjutnya dilakukan tahapan pemeriksaan keterdapatan order *i*' (order sisipan) dan atau order *i* periode berikutnya ($t = t_0 + 1$). Jika keterdapatan order *i*' pada saat periode minggu I berjalan maka dilakukan aturan bahwa perhitungan penjadwalannya dihitung pada periode berikutnya tanpa mengubah jadwal untuk periode minggu I. Apabila keterdapatan order *i* periode berikutnya ($t = t_0 + 1$) maka perhitungan penjadwalannya dihitung pada periode minggu selanjutnya ($t = t_0 + 1$).

Adapun langkah-langkah untuk pemeriksaan keterdapatan order i' (order sisipan) dan atau order i periode berikutnya ($t = t_0 + 1$) adalah sebagai berikut:

- Langkah 0 : PPIC bagian TAB mengambil data order i (buku tulis pada jalur mesin 321) yang akan dijadwalkan (order i dengan $I_i < t_0 + 1$), data order i' (order buku tulis pada jalur mesin 321 susulan)
- Langkah 1 : Tetapkan $t = t_0 + 1$
- Langkah 2 : Periksa keterdapatan order i' pada periode t_0 . Jika ada maka order dihitung kedalam periode $t_0 + 1$. Jika tidak lanjut ke langkah 6
- Langkah 3 : Periksa keterdapatan order i dengan $I_i < t_0 + 1$. Jika ada maka order dihitung kedalam periode $t_0 + 1$. Jika tidak lanjut ke langkah 6
- Langkah 4 : Tulis kembali data order i dan atau order i' yang akan dijadwalkan dalam periode $t_0 + 1$
- Langkah 5 : Kembali hitung data order i dan order i' yang akan dijadwalkan dalam tahap perhitungan *mean lead time* dengan algoritma Zijm-Buitenhek hingga tahap penjadwalan produksi buku tulis
- Langkah 6 : Simpan hasil jadwal produksi buku tulis periode $t_0 + 1$.
Ulangi langkah 0 jika terdapat order i' (order sisipan) dan atau order i periode berikutnya ($t = t_n + 1$). Berhenti

4.3.8 Pengukuran performansi

Tahapan terakhir pada rancangan penjadwalan produksi buku tulis pada jalur mesin 321 yaitu tahapan pengukuran performansi. Pengukuran performansi yang dilakukan pada jumlah *work-in-process (WIP)* dalam produksi buku tulis pada jalur mesin 321 di PT Solo Murni. Perhitungan jumlah *work-in-process (WIP)* diukur dengan *mean flowtime*.

Adapun langkah-langkah untuk tahapan pengukuran performansi adalah sebagai berikut:

- Langkah 0 : PPIC bagian TAB mengambil data order i (buku tulis pada jalur

mesin 321) yang telah dijadwalkan, saat *release* order i urutan ke- k pada *server* s di stasiun kerja j (r_{iks_j}) dan *completion time* order i urutan ke- k pada *server* s di stasiun kerja j (c_{iks_j}).

Langkah 1 : Tulis kembali data order i yang telah dijadwalkan, r_{iks_j} dan c_{iks_j}

Langkah 2 : Hitung *flowtime* sistem keseluruhan dari order i yang telah dijadwalkan dengan pendekatan *DBR* dengan persamaan:

$$F_i = c_{iks_j} - r_{iks_j(j=1)} \quad (\text{jam ke-})$$

Ulang hingga order i ($i=n$) yang telah dijadwalkan selesai dihitung.

Langkah 3 : Hitung *mean flowtime* sistem keseluruhan dengan persamaan:

$$\bar{F} = \frac{1}{n} \sum_{i=1}^n F_i \quad (\text{jam ke-})$$

Langkah 4 : Berhenti

Tersebut diatas adalah uraian mengenai rancangan penjadwalan produksi buku tulis pada jalur mesin 321 di bagian TAB kelompok produk *school supplies* (01). Setelah penjadwalan produksi sudah dilakukan selanjutnya dalam pengendalian produksi dilakukan suatu usaha yang akhirnya pengendalian tersebut berbentuk suatu kebijakan perusahaan. Pengendalian produksi dilakukan karena sebagai pemenuhan fungsi dari Departemen PPIC yang bertugas dalam merencanakan penjadwalan produksi dan mengendalikan produksi jika terdapat suatu penyimpangan dari rencana yang telah dibuat. Kebijakan-kebijakan dalam mengendalikan produksi tersebut yaitu dilakukan lembur dan negosiasi dengan *buyer*.

4.4 Perancangan Program Penjadwalan Produksi Buku Tulis

Pada perancangan program penjadwalan produksi buku tulis ini dilakukan pembuatan konstruksi program. Konstruksi program merupakan penyusunan program penjadwalan produksi yang dibuat. Pada konstruksi program disusun dari langkah-langkah logis hingga dapat memecahkan masalah. Berikut akan dijelaskan perancangan program penjadwalan produksi buku tulis yang dibuat,

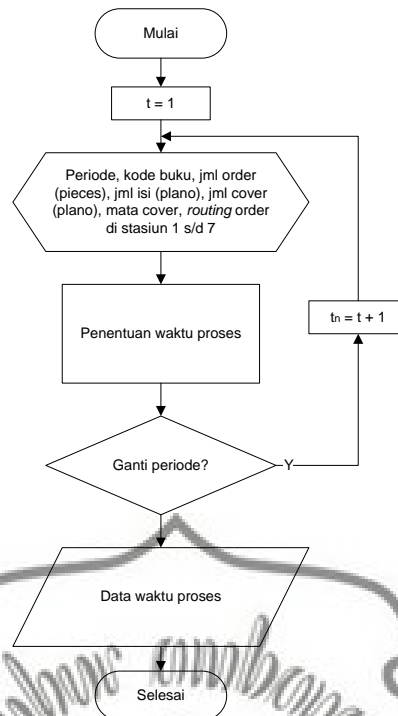
yang meliputi diagram alir program, *syntax* program dan *interface* program yang telah dirancang.

Pada perancangan program penjadwalan produksi buku tulis pada jalur mesin 321 ini terdapat langkah-langkah yang sama seperti pada pengembangan algoritma penjadwalan produksi buku tulis pada jalur mesin 321 diatas. Tahapan tersebut adalah sebagai berikut:

1. Perhitungan *mean lead time* order i di stasiun kerja j (ET_{ij}).
2. Penentuan *earliest bottleneck release* order i ($r_i^{(b)}$) dan *bottleneck due date* order i ($d_i^{(b)}$).
3. Pendistribusian order i ke *server-server* di stasiun kerja *bottleneck* (SKB).
4. Pemeriksaan kelayakan jadwal inisial *server* s di SKB.
5. Revisi jadwal inisial *server* s tidak layak di SKB dengan SPT (*Shortest Processing Time*) untuk meminimasi *mean flowtime*.
6. Penjadwalan produksi di stasiun kerja j .
 - a) Penjadwalan produksi di stasiun kerja *bottleneck*.
 - b) Penjadwalan produksi di stasiun kerja *upstream* (*Backward Scheduling*).
 - c) Penjadwalan produksi di stasiun kerja *downstream* (*Forward Scheduling*).
7. Pengukuran performansi

4.4.1 Penentuan *Mean Lead Time Order* dengan Algoritma Zijm-Buitenhek

Pada tahap awal penentuan penjadwalan produksi buku tulis pada jalur mesin 321 dengan pendekatan *DBR* ini dilakukan penentuan *mean lead time* order dengan menggunakan algoritma Zijm-Buitenhek. Langkah pertama yang dilakukan dalam penentuan *mean lead time* order dengan menggunakan algoritma Zijm-Buitenhek ini yaitu penentuan waktu proses order. Penentuan waktu proses dalam program yang dirancang seperti pada diagram alir program dibawah ini:



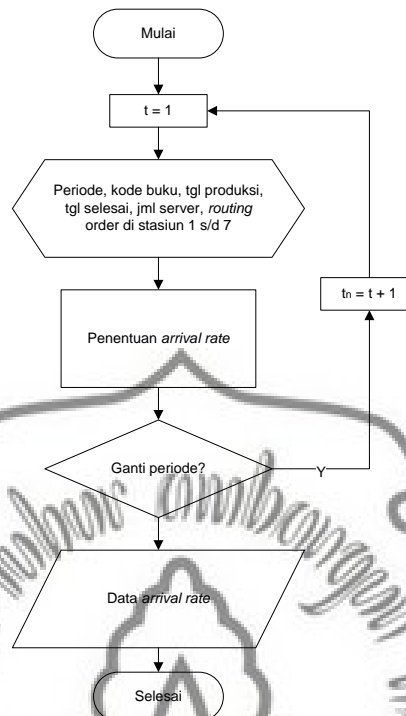
Gambar 4.3 Diagram Alir Untuk Menentukan Waktu Proses Order

Setelah pembuatan algoritma, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan waktu proses order dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program sebagai berikut:

| ROUTING & WAKTU PROSES ORDER | | | | | | | | | | | | | | | | | |
|------------------------------|-----------|--------|--------------------------|--------------------------------|-------|---|---------------------------------|-------|---|-----------------------------------|---|---|---|---|---|---|----------|
| PERIODE ORDER | | MINGGU | | <input type="text" value="1"/> | BULAN | | <input type="text" value="04"/> | TAHUN | | <input type="text" value="2009"/> | | | | | | | |
| DATA ORDER | | | TAHAP Pengerjaan Order I | | | | WAKTU Proses Order I | | | | | | | | | | |
| i | No. Order | Model | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Jam/Unit |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |

Gambar 4.4 *Interface* Program Untuk Menentukan Waktu Proses Order

Pada tahap kedua penentuan *mean lead time* order yaitu penentuan *arrival rate* order. Penentuan *arrival rate* order dalam program yang dirancang seperti pada diagram alir program dibawah ini:

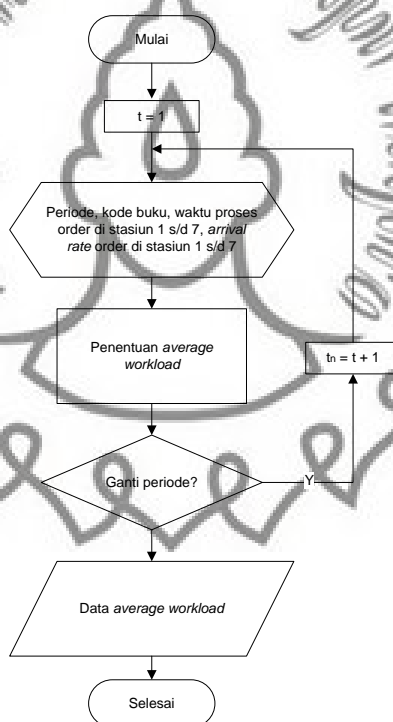


Gambar 4.5 Diagram Alir Untuk Menentukan *Arrival Rate* Order

Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan *arrival rate* order dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program sebagai berikut:

Gambar 4.6 Interface Program Untuk Menentukan *Arrival Rate Order*

Pada tahap ketiga penentuan *mean lead time order* yaitu tahap penentuan *average workload order*. Penentuan *average workload order* dalam program yang dirancang seperti pada diagram alir program dibawah ini:



Gambar 4.7 Diagram Alir Untuk Menentukan *Average Workload Order*

Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan *average workload order* dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini

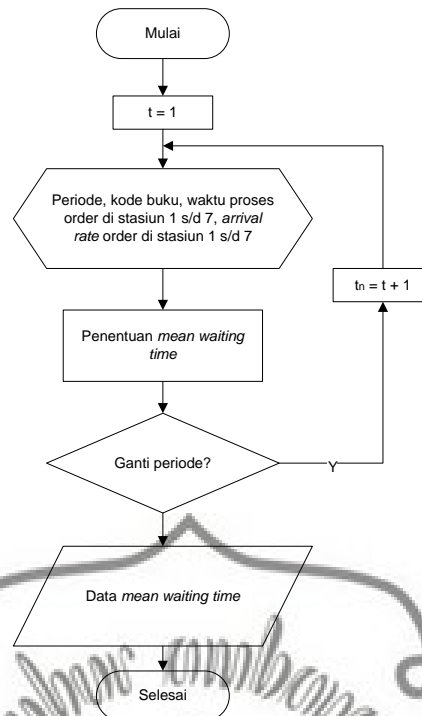
dibuat, kemudian ketika program dijalankan akan muncul *interface* program sebagai berikut:

[illegible]

Gambar 4.8 *Interface Program Untuk Menentukan Average Workload Order*

Pada tahap keempat penentuan *mean lead time order* yaitu tahap penentuan *mean waiting time order*. Penentuan *mean waiting time order* dalam program yang dirancang sesuai diagram alir program seperti pada gambar 4.9.

Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan *mean waiting time* order dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program seperti pada gambar 4.10.

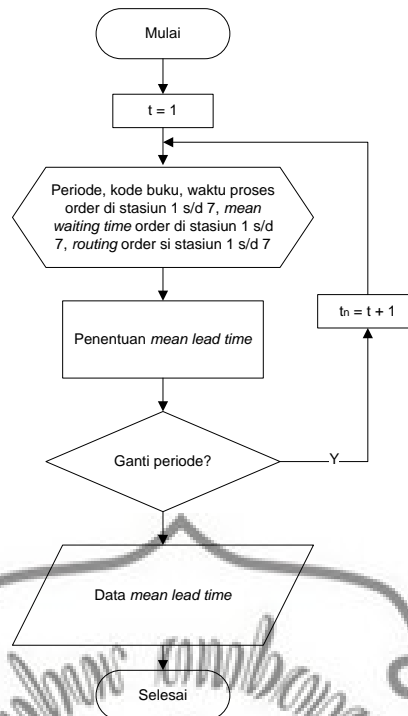


Gambar 4.9 Diagram Alir Untuk Menentukan *Mean Waiting Time Order*

| MEAN WAITING TIME ORDER | | | | | | | | | |
|--------------------------------|-----------|-------|-----------------------------------|---|--------------|---|--------------|---|---|
| PERIODE ORDER | | | MINGGU | | BULAN | | TAHUN | | |
| | | | 1 ▼ | | 04 | | 2009 | | |
| DATA ORDER | | | MEAN WAITING TIME ORDER PERIODE I | | | | | | |
| i | No. Order | Model | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |

Gambar 4.10 *Interface* Program Untuk Menentukan *Mean Waiting Time Order*

Tahap terakhir yaitu tahap penentuan *mean lead time* order. Penentuan *mean lead time* order dalam program yang dirancang seperti pada diagram alir program dibawah ini:



Gambar 4.11 Diagram Alir Untuk Menentukan *Mean Lead Time Order*

Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan *mean lead time* order dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program sebagai berikut:

| MEAN LEAD TIME ORDER | | | | | | | | | | | |
|----------------------|-----------|-------|--------------------------------|--------------------------------|-------|---------------------------------|-------|-----------------------------------|---|----------|--|
| PERIODE ORDER | | | MINGGU | <input type="text" value="1"/> | BULAN | <input type="text" value="04"/> | TAHUN | <input type="text" value="2009"/> | | | |
| DATA ORDER | | | MEAN LEAD TIME ORDER PERIODE I | | | | | | | | |
| i | No. Order | Model | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Jml/Unit | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |

<
 >

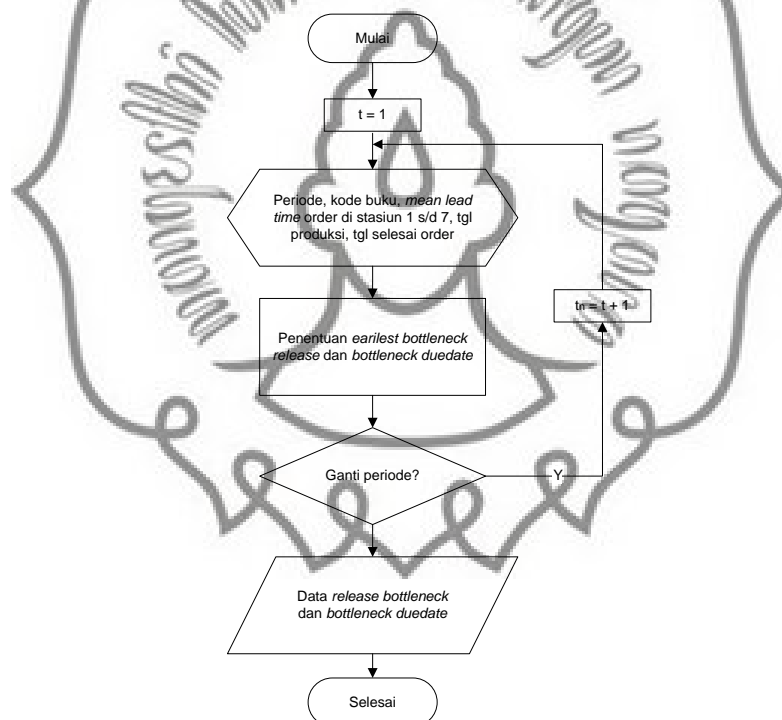
> Proses
 Cetak
 Keluar

Gambar 4.12 *Interface Program Untuk Menentukan Mean Lead Time Order*

4.4.2 Penentuan *Earliest Bottleneck Release Order i* dan *Bottleneck Due Date Order i*

Pada tahap kedua dalam penjadwalan produksi buku tulis pada jalur mesin 321 dengan pendekatan *DBR* ini adalah dilakukan penentuan *earliest bottleneck release order* dan *bottleneck due date order*. Penentuan *earliest bottleneck release order* dan *bottleneck due date order* dalam program yang dirancang sesuai diagram alir program pada gambar 4.13.

Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan *earliest bottleneck release order* dan *bottleneck due date order* dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program seperti pada gambar 4.14.

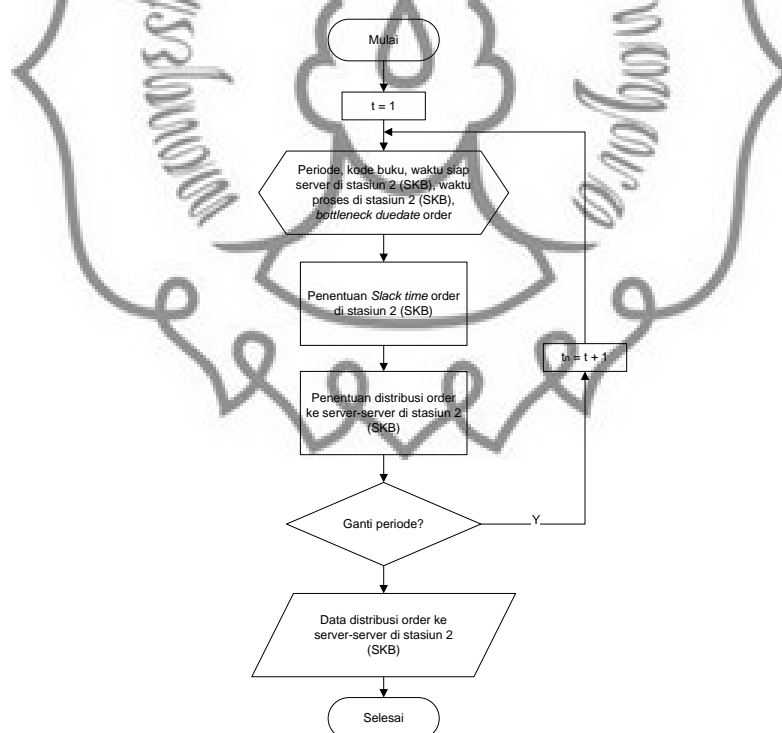


Gambar 4.13 Diagram Alir Untuk Menentukan *Earliest Bottleneck Release Order i* Dan *Bottleneck Due Date Order i*

Gambar 4.14 Interface Program Untuk Menentukan *Earliest Bottleneck Release Order i* Dan *Bottleneck Due Date Order i*

4.4.3 Penentuan Distribusi Order Ke *Server-Server* Di Stasiun *Bottleneck*

Pada tahap ketiga dalam penjadwalan produksi buku tulis pada jalur mesin 321 dengan pendekatan *DBR* ini adalah dilakukan penentuan distribusi order ke *server-server* di stasiun *bottleneck*. Penentuan distribusi order ke *server-server* di stasiun *bottleneck* dalam program yang dirancang seperti pada diagram alir program dibawah ini:



Gambar 4.15 Diagram Alir Untuk Menentukan Distribusi Order Ke *Server-Server* Di Stasiun *Bottleneck*

Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan distribusi order ke *server-server* di

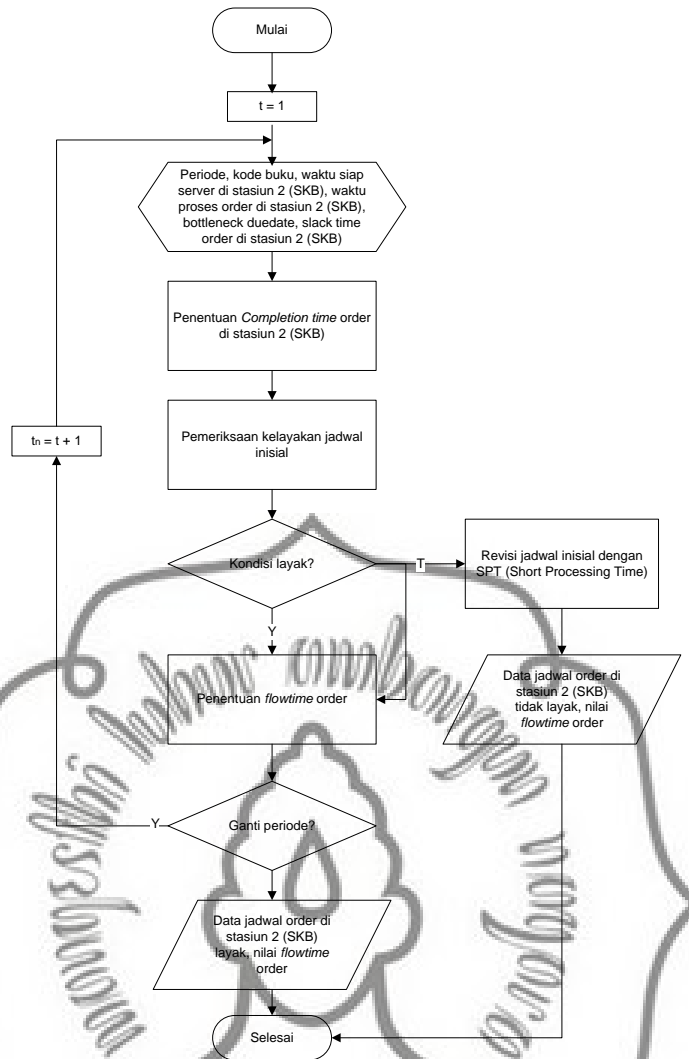
stasiun *bottleneck* dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program sebagai berikut:

Gambar 4.16 *Interface Program Untuk Menentukan Distribusi Order Ke Server-Server Di Stasiun Bottleneck*

4.4.4 Pemeriksaan kelayakan jadwal inisial server s di SKB

Pada tahap keempat dalam penjadwalan produksi buku tulis pada jalur mesin 321 dengan pendekatan *DBR* ini adalah dilakukan pengecekan kelayakan jadwal inisial server di stasiun *bottleneck*. Pengecekan kelayakan jadwal inisial server di stasiun *bottleneck* dalam program yang dirancang sesuai diagram alir program pada gambar 4.17.

Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan pengecekan kelayakan jadwal inisial server di stasiun *bottleneck* dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program seperti pada gambar 4.18.



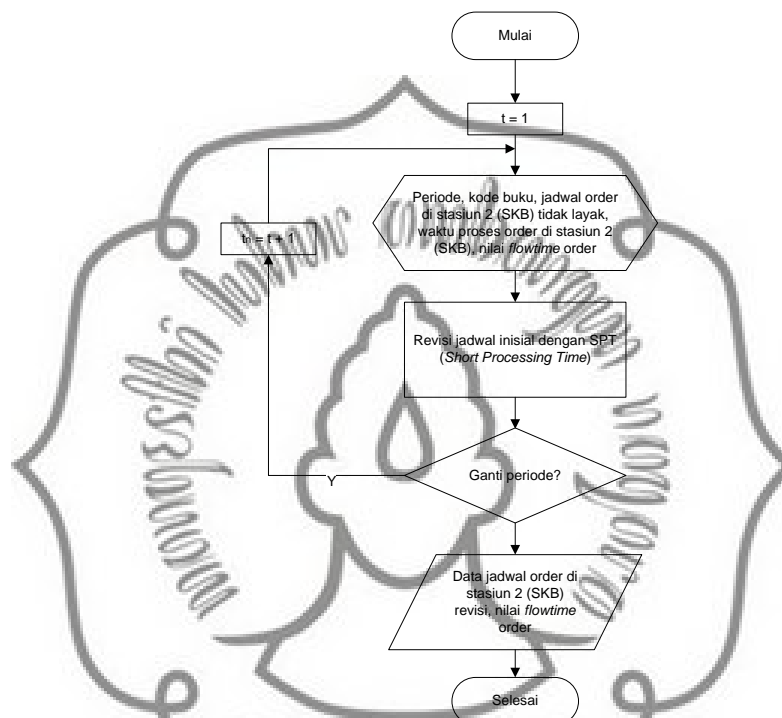
Gambar 4.17 Diagram Alir Untuk Menentukan Pengecekan Kelayakan Jadwal Inisial Server Di Stasiun *Bottleneck*

[illegible]

Gambar 4.18 *Interface* Program Untuk Menentukan Pengecekan Kelayakan Jadwal Inisial Server Di Stasiun *Bottleneck*

4.4.5 Penentuan Revisi Jadwal Inisial Server s Tidak Layak Di Stasiun *Bottleneck* dengan SPT

Pada tahap kelima dalam penjadwalan produksi buku tulis pada jalur mesin 321 dengan pendekatan *DBR* ini adalah dilakukan penentuan revisi jadwal inisial server di stasiun *bottleneck* dengan *SPT*. Revisi ini dilakukan jika kondisi jadwal inisial server di stasiun kerja *bottleneck* memiliki kondisi yang tidak layak. Penentuan revisi jadwal inisial server di stasiun *bottleneck* dengan *SPT* dalam program yang dirancang seperti pada diagram alir program dibawah ini:



Gambar 4.19 Diagram Alir Untuk Menentukan Revisi Jadwal Inisial Server Di Stasiun *Bottleneck*

Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan revisi jadwal inisial server di stasiun *bottleneck* dengan *SPT* dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program sebagai berikut:

Gambar 4.20 *Interface Program Untuk Revisi Jadwal Inisial Server Di Stasiun Bottleneck*

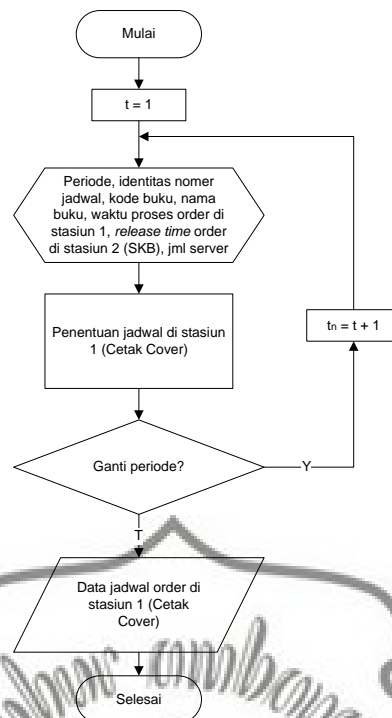
4.4.6 Penentuan Jadwal Produksi Buku Tulis

a. Penjadwalan produksi buku tulis di Stasiun *Bottleneck*

Penjadwalan produksi buku tulis di stasiun *bottleneck* ditentukan dari tahap-tahap sebelumnya, yaitu mulai penentuan *mean lead time* order hingga penentuan pengecekan kelayakan jadwal inisial dan revisi jadwal inisial server di stasiun *bottleneck* jika terdapat kondisi tidak layak pada jadwal inisial tersebut.

b. Penjadwalan produksi di stasiun kerja *upstream* (*Backward Scheduling*)

Setelah hasil jadwal produksi di stasiun kerja *bottleneck* diketahui, tahap selanjutnya dilakukan penjadwalan produksi buku tulis di stasiun kerja *non-bottleneck upstream* yaitu dilakukan penentuan jadwal produksi buku tulis di stasiun 1 (Cetak Cover). Penentuan jadwal produksi buku tulis di stasiun 1 (Cetak Cover) dalam program yang dirancang seperti pada diagram alir program dibawah ini:



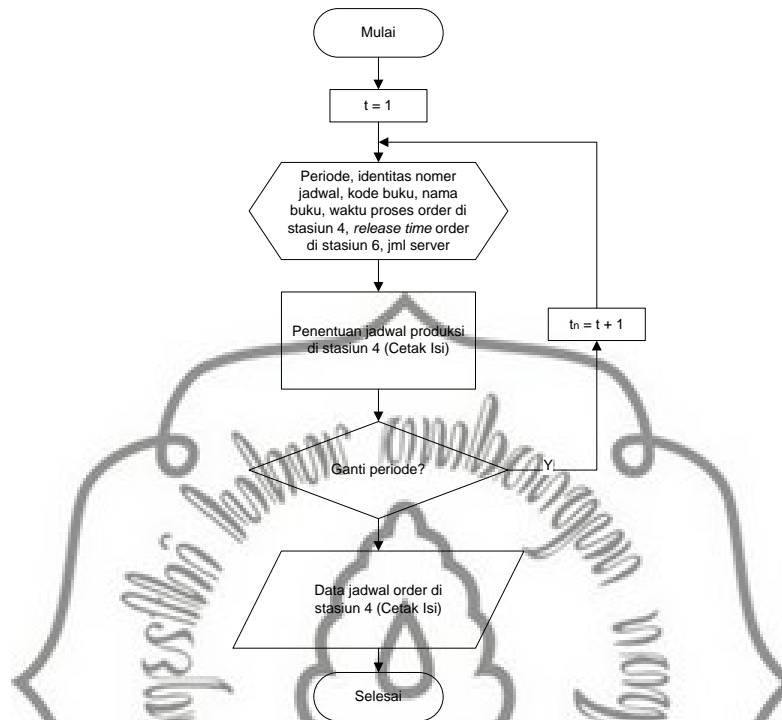
Gambar 4.21 Diagram Alir Untuk Menentukan Jadwal Produksi Buku Tulis Di Stasiun 1 (Cetak Cover)

Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan jadwal produksi buku tulis di stasiun 1 (Cetak Cover) dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program sebagai berikut:

The interface displays a window titled "JADWAL PRODUKSI DI STASIUN KERJA UPSTREAM (CETAK COVER)". It includes a header section with fields for "PERIODE ORDER", "MINGGU" (with a dropdown), "BULAN" (set to 04), and "TAHUN" (set to 2009). Below this is a table with the following columns: "i", "No. Order", "Model", "Riksj (+1)", "Pij", "Riksj", "Ciksj", "Server", and "Urutan". The table contains multiple empty rows for data entry. At the bottom of the window, there are three buttons: "Proses", "Cetak", and "Keluar".

Gambar 4.22 Interface Program Untuk Menentukan Jadwal Produksi Buku Tulis Di Stasiun 1 (Cetak Cover)

Selanjutnya adalah penjadwalan untuk stasiun kerja *non-bottleneck upstream* yaitu jadwal produksi buku tulis di stasiun 4 (Cetak Isi). Penentuan jadwal produksi buku tulis di stasiun 4 (Cetak Isi) dalam program yang dirancang seperti pada diagram alir program dibawah ini:



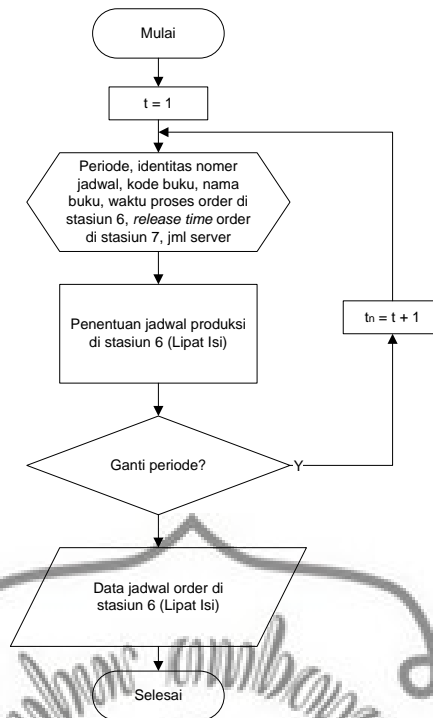
Gambar 4.23 Diagram Alir Untuk Menentukan Jadwal Produksi Buku Tulis Di Stasiun 4 (Cetak Isi)

Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan jadwal produksi buku tulis di stasiun 4 (Cetak Isi) dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program sebagai berikut:

Gambar 4.24 Interface Program Untuk Menentukan Jadwal Produksi Buku Tulis Di Stasiun 4 (Cetak Isi)

Selanjutnya dilakukan penjadwalan untuk stasiun kerja *non-bottleneck upstream* yaitu jadwal produksi buku tulis di stasiun 6 (Lipat Isi). Penentuan jadwal produksi buku tulis di stasiun 6 (Lipat Isi) dalam program yang dirancang sesuai diagram alir program pada gambar 4.25.

Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan jadwal produksi buku tulis di stasiun 6 (Lipat Isi) dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program seperti pada gambar 4.26.



Gambar 4.25 Diagram Alir Untuk Menentukan Jadwal Produksi Buku Tulis Di Stasiun 6 (Lipat Isi)

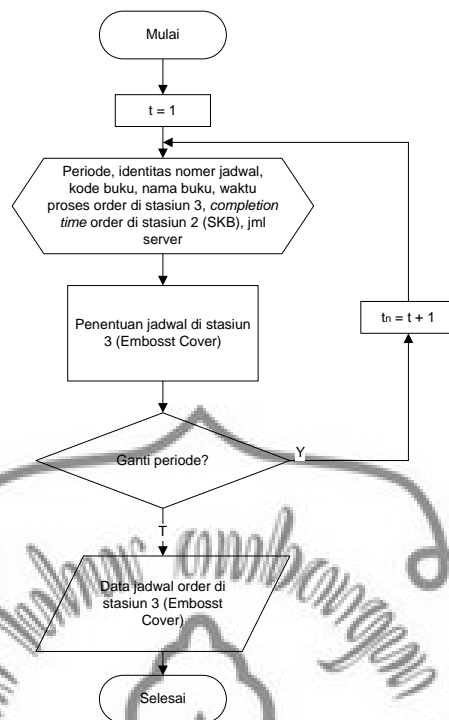
The interface shows a window titled "JADWAL PRODUKSI DI STASIUN KERJA UPSTREAM (LIPAT ISI)". It includes input fields for "PERIODE ORDER", "MINGGU" (with a dropdown), "BULAN" (set to 04), and "TAHUN" (set to 2009). Below these is a table with columns: I, No. Order, Model, Riks_j (j+1), Pij, Riks_j, Ciks_j, Server, and Urutan. The table has 15 empty rows. At the bottom, there are buttons for "Proses", "Cetak", and "Keluar".

Gambar 4.26 Interface Program Untuk Menentukan Jadwal Produksi Buku Tulis Di Stasiun 6 (Lipat Isi)

c. Penjadwalan produksi di stasiun kerja *downstream* (Forward Scheduling)

Pada tahap selanjutnya dilakukan penjadwalan produksi buku tulis untuk stasiun kerja *non-bottleneck downstream* yaitu dilakukan penentuan jadwal produksi buku tulis di stasiun 3 (Embosst Cover). Penentuan jadwal produksi

buku tulis di stasiun 3 (Embosst Cover) dalam program yang dirancang seperti pada diagram alir program dibawah ini:



Gambar 4.27 Diagram Alir Untuk Menentukan Jadwal Produksi Buku Tulis Di Stasiun 3 (Embosst Cover)

Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan jadwal produksi buku tulis di stasiun 3 (Embosst Cover) dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program sebagai berikut:

| JADWAL PRODUKSI DI STASIUN KERJA DOWNSTREAM (EMBOSS COVER) | | | | | | | | | |
|---|-----------|-------|---|-----|---|-------|---|--------|--|
| PERIODE ORDER | | | MINGGU ▼ | | BULAN 04 | | TAHUN 2009 | | |
| i | No. Order | Model | Ciksj (j-1) | Pij | Riksj | Ciksj | Server | Urutan | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |

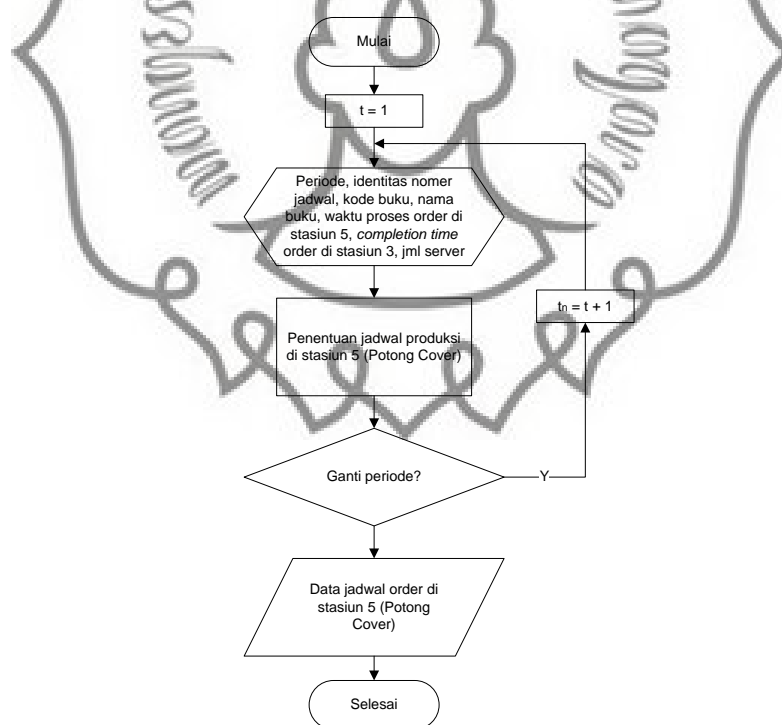
<
>

➤ Proses
🖨️ Cetak

✕ Keluar

Gambar 4.28 *Interface* Program Untuk Menentukan Jadwal Produksi Buku Tulis Di Stasiun 3 (Embosst Cover)

Pada tahap selanjutnya dilakukan penentuan jadwal produksi buku tulis di stasiun 5 (Potong Cover). Penentuan jadwal produksi buku tulis di stasiun 5 (Potong Cover) dalam program yang dirancang seperti pada diagram alir program dibawah ini:



Gambar 4.29 Diagram Alir Untuk Menentukan Jadwal Produksi Buku Tulis Di Stasiun 5 (Potong Cover)

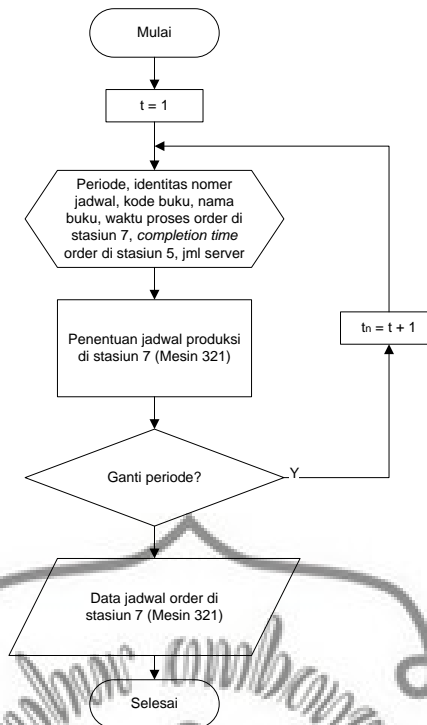
Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan jadwal produksi buku tulis di stasiun 5

(Potong Cover) dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program sebagai berikut:

Gambar 4.30 *Interface* Program Untuk Menentukan Jadwal Produksi Buku Tulis Di Stasiun 5 (Potong Cover)

Pada tahap selanjutnya dilakukan penentuan jadwal produksi buku tulis di stasiun 7 (Mesin 321). Penentuan jadwal produksi buku tulis di stasiun 7 (Mesin 321) dalam program yang dirancang sesuai diagram alir program pada gambar 4.31.

Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan jadwal produksi buku tulis di stasiun 7 (Mesin 321) dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program seperti pada gambar 4.32.



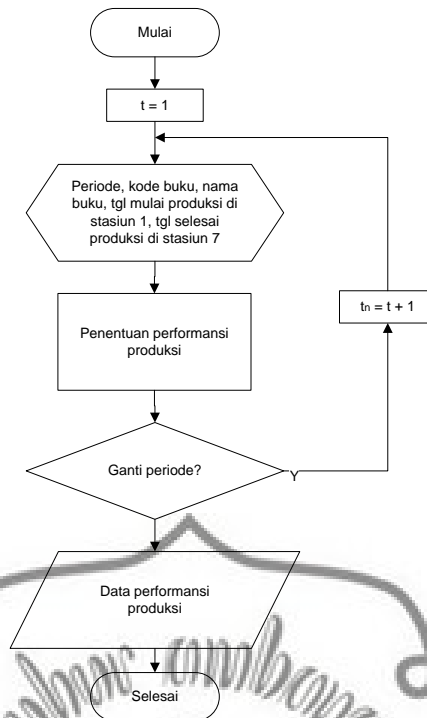
Gambar 4.31 Diagram Alir Untuk Menentukan Jadwal Produksi Buku Tulis Di Stasiun 7 (Mesin 321)

The interface shows a window titled "JADWAL PRODUKSI DI STASIUN KERJA DOWNSTREAM [MESIN 321]". It includes a header section with "PERIODE ORDER", "MINGGU" (with a dropdown), "BULAN" (04), and "TAHUN" (2009). Below this is a table with columns: i, No. Order, Model, Ciks (j-1), Pij, Riksj, Ciks, Server, and Urutan. The table is currently empty. At the bottom, there are buttons for "Proses", "Cetak", and "Keluar".

Gambar 4.32 Interface Program Untuk Menentukan Jadwal Produksi Buku Tulis Di Stasiun 7 (Mesin 321)

4.4.7 Penentuan Performansi Jadwal Produksi Buku Tulis

Pada tahap terakhir dilakukan penentuan performansi jadwal produksi buku tulis di jalur mesin 321. Penentuan performansi jadwal produksi buku tulis di jalur mesin 321 dalam program yang dirancang seperti pada diagram alir program dibawah ini:



Gambar 4.33 Diagram Alir Untuk Menentukan Performansi Jadwal Produksi Buku Tulisdi Jahur Mesin 321

Setelah pembuatan algoritma tersebut, kemudian diterjemahkan dalam kode program (*syntax*) sehingga penentuan performansi jadwal produksi buku tulisi jalur mesin 321 dapat dilakukan. Kode program tersebut merupakan kode program yang diterjemahkan dalam pemrograman komputer dengan Visual FoxPro. Setelah kode program ini dibuat, kemudian ketika program dijalankan akan muncul *interface* program sebagai berikut:

[illegible]

Gambar 4.34 *Interface* Program Untuk Menentukan Performansi Jadwal Produksi Buku Tulis di Jalur Mesin 321

4.5 Validasi Program Penjadwalan Produksi Buku Tulis

4.5.1 Validasi Algoritma Penjadwalan Produksi Buku Tulis Dengan Pendekatan *DBR*

Sebelum melakukan validasi program penjadwalan produksi buku tulis, langkah yang dilakukan yaitu memvalidasi algoritma penjadwalan produksi buku tulis yang melewati jalur mesin 321 dengan menggunakan pendekatan *DBR*. Validasi algoritma ini dengan menggunakan kriteria performansi (Costatine dan Ulvila, 1990) yaitu dengan penilaian (*judgement*). Atribut yang dinilai pada *judgment* ini adalah kualitas dari alasan, dimana penilaian pengguna terhadap kesesuaian jawaban yang diberikan oleh sistem terhadap permasalahan yang diberikan. Penilaian tersebut dilakukan oleh manager PPIC di PT. Solo Murni. Kriteria validasi dari *Judgment* (penilaian), yaitu: Apakah algoritma penjadwalan produksi buku tulis dengan menggunakan *DBR* dapat diterapkan di PT. Solo Murni?

Menurut manager PPIC di PT. Solo Murni, bahwa algoritma penjadwalan produksi buku tulis yang melewati jalur mesin 321 dengan menggunakan pendekatan *DBR* dapat diterapkan di PT. Solo Murni. Hal ini karena di lini produksi buku tulis pada jalur mesin 321 di PT. Solo Murni terdapat sistem *bottleneck* yaitu terdapat stasiun kerja yang memiliki kapasitas kecil atau terbatas yaitu pada Vurnish Cover. Pada Vurnish Cover tersebut hanya memiliki satu unit mesin. Selain itu sistem penjadwalan produksi yang diterapkan di PT. Solo Murni menggunakan penjadwalan maju (*forward scheduling*).

4.5.2 Validasi Program Aplikasi Penjadwalan Produksi Buku Tulis

Setelah pengujian validasi algoritma penjadwalan produksi buku tulis yang melewati jalur mesin 321 dengan menggunakan pendekatan *DBR*, maka selanjutnya dilakukan validasi program aplikasi penjadwalan produksi buku tulis. Validasi program aplikasi yaitu membandingkan antara hasil penjadwalan produksi buku tulis dengan pendekatan *DBR* yang di simulasikan dengan perhitungan manual dan hasil penjadwalan produksi buku tulis dengan pendekatan *DBR* yang di simulasikan dengan program aplikasi yang telah disusun.

Pada proses validasi ini, diambil contoh perhitungan untuk satu periode produksi buku tulis yaitu pada data periode pertama bulan April 2009. Contoh tersebut dihitung menurut tahapan penjadwalan produksi buku tulis dengan menggunakan *DBR* sebagai berikut:

1. Perhitungan *mean lead time* order i di stasiun kerja j (ET_{ij}).
2. Penentuan *earliest bottleneck release* order i ($r_i^{(b)}$) dan *bottleneck due date* order i ($d_i^{(b)}$).
3. Pendistribusian order i ke *server-server* di stasiun kerja *bottleneck* (SKB).
4. Pemeriksaan kelayakan jadwal inisial *server* s di SKB.
5. Revisi jadwal inisial *server* s tidak layak di SKB dengan SPT (*Shortest Processing Time*) untuk meminimasi *mean flowtime*.
6. Penjadwalan produksi di stasiun kerja j .
 - a) Penjadwalan produksi di stasiun kerja *bottleneck*.
 - b) Penjadwalan produksi di stasiun kerja *upstream* (*Backward Scheduling*).
 - c) Penjadwalan produksi di stasiun kerja *downstream* (*Forward Scheduling*).
7. Pengukuran performansi

Adapun hasil perbandingan perhitungan manual dan hasil yang disimulasikan dengan program aplikasi adalah sebagai berikut:

A. Penentuan *Mean Lead Time* Order dengan Algoritma Zijm-Buitenhek

Pada tahap awal proses validasi program aplikasi penjadwalan produksi buku tulis pada jalur mesin 321 dengan menggunakan pendekatan *DBR* ini dilakukan validasi proses penentuan *mean lead time* order dengan menggunakan algoritma Zijm-Buitenhek pada program aplikasi. Dalam memvalidasi proses penentuan *mean lead time* order tersebut, langkah pertama yang dilakukan yaitu membandingkan hasil waktu proses order yang dihitung secara manual dan hasil waktu proses order yang dihitung dengan program aplikasi yang telah dirancang. Hasil perhitungan tersebut adalah sebagai berikut:

Tabel 4.4 Waktu Proses Order Periode I (Manual)

| Waktu Proses Order-order (pij) Periode I | | | | | | | | | | |
|--|-----------|----------------------------------|-----------------|---------|--------|---------|--------|---------|---------|--------------|
| Order i | No. Order | Model | Stasiun Kerja j | | | | | | | Jumlah (jam) |
| | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 5.4501 | 12.2404 | 4.6784 | 11.5862 | 0.8547 | 11.9893 | 11.7895 | 58.5886 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 5.4501 | 12.2404 | 4.6784 | 11.5862 | 0.8547 | 11.9893 | 11.7895 | 58.5886 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 5.4501 | 12.2404 | 4.6784 | 11.5862 | 0.8547 | 11.9893 | 11.7895 | 58.5886 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 5.4501 | 12.2404 | 4.6784 | 11.5862 | 0.8547 | 11.9893 | 11.7895 | 58.5886 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 0.3406 | 0.7650 | 0.2924 | 0.7241 | 0.0534 | 0.7493 | 0.7368 | 3.6618 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 2.7251 | 6.1202 | 2.3392 | 3.7655 | 0.4274 | 3.8965 | 5.8947 | 25.1686 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 5.4501 | 12.2404 | 4.6784 | 7.5310 | 0.8547 | 7.7930 | 11.7895 | 50.3372 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 1.0900 | 2.4481 | 0.9357 | 2.3172 | 0.1709 | 2.3979 | 2.3579 | 11.7177 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 1.6350 | 3.6721 | 1.4035 | 3.4759 | 0.2564 | 3.5968 | 3.5368 | 17.5766 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 6.1314 | 13.7705 | 5.2632 | 2.8966 | 0.9615 | 2.9973 | 5.8947 | 37.9152 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 5.5182 | 12.3934 | 4.7368 | 2.6069 | 0.8654 | 2.6976 | 10.6105 | 39.4289 |

| ROUTING & WAKTU PROSES ORDER | | | | | | | | | | | | | | | | | |
|---|-----------|---------------------------------|---|---------------|---|--------------|-----------------------------|--------------|------|--------|---------|--------|---------|--------|---------|---------|----------|
| | | PERIODE ORDER | | MINGGU | 1 | BULAN | 04 | TAHUN | 2009 | | | | | | | | |
| DATA ORDER | | TAHAJ PENGERJAAN ORDER I | | | | | WAKTU PROSES ORDER I | | | | | | | | | | |
| i | No. Order | Model | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Jam/Unit |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 5.3949 | 12.1170 | 4.6326 | 11.5857 | 0.8463 | 11.9889 | 11.7895 | 58.3549 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 5.3949 | 12.1170 | 4.6326 | 11.5857 | 0.8463 | 11.9889 | 11.7895 | 58.3549 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 5.3949 | 12.1170 | 4.6326 | 11.5857 | 0.8463 | 11.9889 | 11.7895 | 58.3549 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 5.3949 | 12.1170 | 4.6326 | 11.5857 | 0.8463 | 11.9889 | 11.7895 | 58.3549 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOM | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.3381 | 0.7581 | 0.2898 | 0.7245 | 0.0525 | 0.7497 | 0.7368 | 3.6495 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2.6985 | 6.0585 | 2.3163 | 3.7653 | 0.4221 | 3.8955 | 5.8947 | 25.0509 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 5.3949 | 12.1170 | 4.6326 | 7.5306 | 0.8463 | 7.7931 | 11.7895 | 50.1040 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1.0794 | 2.4234 | 0.9261 | 2.3163 | 0.1701 | 2.3982 | 2.3579 | 11.6714 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1.6191 | 3.6351 | 1.3902 | 3.4755 | 0.2541 | 3.5973 | 3.5368 | 17.5081 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 6.1320 | 13.7697 | 5.2626 | 2.8959 | 0.9618 | 2.9967 | 5.8947 | 37.9134 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 5.7393 | 12.8898 | 4.9266 | 2.6061 | 0.9009 | 2.6985 | 10.6105 | 40.3717 |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |

Proses Cetak Keluar

Gambar 4.35 Waktu Proses Order Periode I (Program Aplikasi)

Validasi program aplikasi pada tahap kedua yaitu membandingkan hasil *arrival rate* order yang dihitung secara manual dan hasil *arrival rate* order yang dihitung dengan program aplikasi yang telah dirancang. Hasil perhitungan tersebut adalah sebagai berikut:

Tabel 4.5 *Arrival Rate* Order Periode I (Manual)

| Arrival Rate (λ_{ij}) (unit/jam) | | | | | | | | | |
|--|-----------|----------------------------------|-----------------|--------|--------|--------|--------|--------|--------|
| Order i | No. Order | Model | Stasiun Kerja j | | | | | | |
| | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 0.0060 | 0.0060 | 0.0020 | 0.0010 | 0.0030 | 0.0012 | 0.0020 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 0.0060 | 0.0060 | 0.0020 | 0.0010 | 0.0030 | 0.0012 | 0.0020 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 0.0060 | 0.0060 | 0.0020 | 0.0010 | 0.0030 | 0.0012 | 0.0020 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 0.0030 | 0.0030 | 0.0010 | 0.0005 | 0.0015 | 0.0006 | 0.0010 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 0.0014 | 0.0014 | 0.0005 | 0.0002 | 0.0007 | 0.0003 | 0.0005 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 0.0008 | 0.0008 | 0.0003 | 0.0001 | 0.0004 | 0.0002 | 0.0003 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 0.0014 | 0.0014 | 0.0005 | 0.0002 | 0.0007 | 0.0003 | 0.0005 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 0.0021 | 0.0021 | 0.0007 | 0.0003 | 0.0010 | 0.0004 | 0.0007 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 0.0021 | 0.0021 | 0.0007 | 0.0003 | 0.0010 | 0.0004 | 0.0007 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 0.0014 | 0.0014 | 0.0005 | 0.0002 | 0.0007 | 0.0003 | 0.0005 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 0.0014 | 0.0014 | 0.0005 | 0.0002 | 0.0007 | 0.0003 | 0.0005 |

ARRIVAL RATE ORDER

PERIODE ORDER MINGGU 1 BULAN 04 TAHUN 2009

| DATA ORDER | | | | | | | ARRIVAL RATE ORDER | | | | | | | |
|------------|-----------|--------------------------------|------------|--------------|---------------|------|--------------------|--------|--------|--------|--------|--------|--------|--------|
| i | No. Order | Model | Tgl. Order | Tgl. Selesai | Tgl. Produksi | PLT | Di | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS | 08-04-2009 | 21-04-2009 | 13-04-2009 | 168 | 0.0060 | 0.0060 | 0.0060 | 0.0020 | 0.0010 | 0.0030 | 0.0012 | 0.0020 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS | 08-04-2009 | 21-04-2009 | 13-04-2009 | 168 | 0.0060 | 0.0060 | 0.0060 | 0.0020 | 0.0010 | 0.0030 | 0.0012 | 0.0020 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS | 08-04-2009 | 21-04-2009 | 13-04-2009 | 168 | 0.0060 | 0.0060 | 0.0060 | 0.0020 | 0.0010 | 0.0030 | 0.0012 | 0.0020 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS | 08-04-2009 | 30-04-2009 | 13-04-2009 | 336 | 0.0030 | 0.0030 | 0.0030 | 0.0010 | 0.0005 | 0.0015 | 0.0006 | 0.0010 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOM | 08-04-2009 | 23-05-2009 | 13-04-2009 | 714 | 0.0014 | 0.0014 | 0.0014 | 0.0005 | 0.0002 | 0.0007 | 0.0003 | 0.0005 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA | 10-04-2009 | 24-06-2009 | 13-04-2009 | 1281 | 0.0008 | 0.0008 | 0.0008 | 0.0003 | 0.0001 | 0.0004 | 0.0002 | 0.0003 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA | 10-04-2009 | 24-05-2009 | 13-04-2009 | 714 | 0.0014 | 0.0014 | 0.0014 | 0.0005 | 0.0002 | 0.0007 | 0.0003 | 0.0005 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS | 10-04-2009 | 10-05-2009 | 13-04-2009 | 483 | 0.0021 | 0.0021 | 0.0021 | 0.0007 | 0.0004 | 0.0011 | 0.0004 | 0.0007 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS | 10-04-2009 | 10-05-2009 | 13-04-2009 | 483 | 0.0021 | 0.0021 | 0.0021 | 0.0007 | 0.0004 | 0.0011 | 0.0004 | 0.0007 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 10-04-2009 | 23-05-2009 | 13-04-2009 | 714 | 0.0014 | 0.0014 | 0.0014 | 0.0005 | 0.0002 | 0.0007 | 0.0003 | 0.0005 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 10-04-2009 | 22-05-2009 | 13-04-2009 | 693 | 0.0014 | 0.0014 | 0.0014 | 0.0005 | 0.0002 | 0.0007 | 0.0003 | 0.0005 |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |

<

Gambar 4.37 Average Workload Order Periode I (Program Aplikasi)

Validasi program aplikasi pada tahap keempat yaitu membandingkan hasil *mean waiting time* order yang dihitung secara manual dan hasil *mean waiting time* order yang dihitung dengan program aplikasi yang telah dirancang. Hasil tersebut adalah sebagai berikut:

Tabel 4.7 Mean Waiting Time Order Periode I (Manual)

| Mean Waiting Time (W _j) Periode I | | | | | | | | | |
|---|-----------|----------------------------------|-----------------|--------|--------|--------|--------|--------|--------|
| Order i | No. Order | Model | Stasiun Kerja j | | | | | | |
| | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 0.1768 | 0.8918 | 0.0434 | 0.1332 | 0.0022 | 0.1711 | 0.2758 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 0.1768 | 0.8918 | 0.0434 | 0.1332 | 0.0022 | 0.1711 | 0.2758 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 0.1768 | 0.8918 | 0.0434 | 0.1332 | 0.0022 | 0.1711 | 0.2758 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 0.0884 | 0.4459 | 0.0217 | 0.0666 | 0.0011 | 0.0856 | 0.1379 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 0.0002 | 0.0008 | 0.0000 | 0.0001 | 0.0000 | 0.0002 | 0.0003 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 0.0058 | 0.0292 | 0.0014 | 0.0018 | 0.0001 | 0.0024 | 0.0090 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 0.0416 | 0.2098 | 0.0102 | 0.0132 | 0.0005 | 0.0170 | 0.0649 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 0.0025 | 0.0124 | 0.0006 | 0.0019 | 0.0000 | 0.0024 | 0.0038 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 0.0055 | 0.0279 | 0.0014 | 0.0042 | 0.0001 | 0.0054 | 0.0086 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 0.0527 | 0.2656 | 0.0129 | 0.0020 | 0.0006 | 0.0025 | 0.0162 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 0.0439 | 0.2216 | 0.0108 | 0.0016 | 0.0005 | 0.0021 | 0.0542 |

<

Gambar 4.38 Mean Waiting Time Order Periode I (Program Aplikasi)

Tahap terakhir yaitu memvalidasi pada proses penentuan *mean lead time* order. Validasi program aplikasi tersebut dilakukan dengan membandingkan hasil *mean lead time* order yang dihitung secara manual dan hasil *mean lead time* order yang dihitung dengan program aplikasi yang telah dirancang. Hasil tersebut adalah sebagai berikut:

Tabel 4.8 Mean Lead Time Order Periode I (Manual)

| Mean Lead Time (ETij) Periode I | | | | | | | | | | |
|---------------------------------|-----------|----------------------------------|-----------------|---------|--------|---------|--------|---------|---------|-------------------|
| Order i | No. Order | Model | Stasiun Kerja j | | | | | | | Jumlah (jam/unit) |
| | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 14.2446 | 4.7734 | 11.8327 | 0.8594 | 12.3064 | 12.3558 | 62.2131 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 14.2446 | 4.7734 | 11.8327 | 0.8594 | 12.3064 | 12.3558 | 62.2131 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 14.2446 | 4.7734 | 11.8327 | 0.8594 | 12.3064 | 12.3558 | 62.2131 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 14.2446 | 4.7734 | 11.8327 | 0.8594 | 12.3064 | 12.3558 | 62.2131 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 0.7313 | 2.7692 | 0.3874 | 0.9706 | 0.0582 | 1.0664 | 1.3032 | 7.2863 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 3.1157 | 8.1244 | 2.4342 | 4.0120 | 0.4321 | 4.2136 | 6.4611 | 28.7931 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 5.8408 | 14.2446 | 4.7734 | 7.7775 | 0.8594 | 8.1101 | 12.3558 | 53.9617 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 1.4807 | 4.4523 | 1.0307 | 2.5637 | 0.1757 | 2.7149 | 2.9242 | 15.3422 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 2.0257 | 5.6763 | 1.4986 | 3.7224 | 0.2612 | 3.9138 | 4.1032 | 21.2011 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 6.5221 | 15.7747 | 5.3582 | 3.1430 | 0.9663 | 3.3144 | 6.4611 | 41.5397 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 5.9089 | 14.3976 | 4.8319 | 2.8534 | 0.8701 | 3.0146 | 11.1769 | 43.0535 |

| i | No. Order | Model | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Jml/Unit |
|----|-----------|--------------------------------|--------|---------|--------|---------|--------|---------|---------|----------|
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS | 5.7822 | 14.1045 | 4.7282 | 11.8330 | 0.8510 | 12.3093 | 12.3646 | 61.9728 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS | 5.7822 | 14.1045 | 4.7282 | 11.8330 | 0.8510 | 12.3093 | 12.3646 | 61.9728 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS | 5.7822 | 14.1045 | 4.7282 | 11.8330 | 0.8510 | 12.3093 | 12.3646 | 61.9728 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS | 5.7822 | 14.1045 | 4.7282 | 11.8330 | 0.8510 | 12.3093 | 12.3646 | 61.9728 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOM | 0.7254 | 2.7456 | 0.3854 | 0.9718 | 0.0572 | 1.0701 | 1.3119 | 7.2674 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA | 3.0858 | 8.0460 | 2.4119 | 4.0126 | 0.4268 | 4.2159 | 6.4698 | 28.6688 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA | 5.7822 | 14.1045 | 4.7282 | 7.7779 | 0.8510 | 8.1135 | 12.3646 | 53.7219 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS | 1.4667 | 4.4109 | 1.0217 | 2.5636 | 0.1748 | 2.7186 | 2.9330 | 15.2893 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS | 2.0064 | 5.6226 | 1.4858 | 3.7228 | 0.2588 | 3.9177 | 4.1119 | 21.1260 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 6.5193 | 15.7572 | 5.3582 | 3.1432 | 0.9665 | 3.3171 | 6.4698 | 41.5313 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 6.1266 | 14.8773 | 5.0222 | 2.8534 | 0.9056 | 3.0189 | 11.1856 | 43.9896 |

Gambar 4.39 Mean Lead Time Order Periode (Program Aplikasi)

B. Penentuan *Earliest Bottleneck Release Order i* dan *Bottleneck Due Date Order i*

Pada tahap kedua dalam validasi program aplikasi penjadwalan produksi buku tulis pada jalur mesin 321 dengan pendekatan *DBR* ini adalah dilakukan validasi proses penentuan *earliest bottleneck release order* dan *bottleneck due date order* pada program aplikasi. Validasi proses penentuan *earliest bottleneck release order* dan *bottleneck due date order* yaitu dengan membandingkan hasil antara perhitungan manual dan perhitungan dengan program aplikasi. Hasil tersebut adalah sebagai berikut:

Tabel 4.9 *Earliest Bottleneck Release Order i dan Bottleneck Due Date Order i*
(Manual)

| Earliest Bottleneck Release (rib) & Bottleneck Due Date (dib) Periode I | | | | | | | | | |
|---|-----------|----------------------------------|------------------------------|----------------|-------|---------|--|--|--|
| Order i | No. Order | Model | Stasiun Kerja 2 (Bottleneck) | | | Kondisi | | | |
| | | | rib (jam ke-) | dib (jam ke-) | | | | | |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 150.0113 | Layak | | | | |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 150.0113 | Layak | | | | |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 150.0113 | Layak | | | | |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 318.0113 | Layak | | | | |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 0.7313 | 712.2512 | Layak | | | | |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 3.1157 | 1271.6726 | Layak | | | | |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 5.8408 | 696.0113 | Layak | | | | |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 1.4807 | 478.8694 | Layak | | | | |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 2.0257 | 477.1371 | Layak | | | | |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 6.5221 | 701.2144 | Layak | | | | |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 5.9089 | 676.1211 | Layak | | | | |

Gambar 4.40 *Earliest Bottleneck Release Order dan Bottleneck Due Date Order*
(Program Aplikasi)

C. Penentuan Distribusi Order Ke Server-Server Di Stasiun Bottleneck

Pada tahap ketiga dalam validasi program aplikasi penjadwalan produksi buku tulis pada jalur mesin 321 dengan pendekatan *DBR* ini adalah dilakukan validasi proses penentuan distribusi order ke *server-server* di stasiun *bottleneck*. Validasi proses penentuan distribusi order ke *server-server* di stasiun *bottleneck* yaitu dengan membandingkan hasil antara perhitungan manual dan perhitungan dengan program aplikasi. Hasil tersebut adalah sebagai berikut:

Tabel 4.10 *Distribusi Order Ke Server-Server Di Stasiun Bottleneck*
(Manual)

| Order i | No. Order | Model | rib (jam ke-) | dib (jam ke-) | pij | STib | Urutan distrb. |
|---------|-----------|----------------------------------|----------------|----------------|---------|-----------|----------------|
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 150.0113 | 12.2404 | 131.7709 | 1 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 150.0113 | 12.2404 | 131.7709 | 2 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 150.0113 | 12.2404 | 131.7709 | 3 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 318.0113 | 12.2404 | 299.7709 | 4 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 2.0257 | 477.1371 | 3.6721 | 467.4650 | 5 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 1.4807 | 478.8694 | 2.4481 | 470.4213 | 6 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 5.9089 | 676.1211 | 12.3934 | 657.7277 | 7 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 5.8408 | 696.0113 | 12.2404 | 677.7709 | 8 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 6.5221 | 701.2144 | 13.7705 | 681.4440 | 9 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 0.7313 | 712.2512 | 0.7650 | 705.4862 | 10 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 3.1157 | 1271.6726 | 6.1202 | 1259.5524 | 11 |

[illegible]

Gambar 4.41 Distribusi Order Ke *Server-Server* Di Stasiun *Bottleneck* (Program Aplikasi)

D. Pengecekan Kelayakan Jadwal Inisial Server Di Stasiun *Bottleneck*

Pada tahap keempat dalam validasi program aplikasi penjadwalan produksi buku tulis pada jalur mesin 321 dengan pendekatan *DBR* ini adalah dilakukan validasi proses pengecekan kelayakan jadwal inisial server di stasiun *bottleneck*. Validasi proses pengecekan kelayakan jadwal inisial server di stasiun *bottleneck* yaitu dengan membandingkan hasil antara perhitungan manual dan perhitungan dengan program aplikasi. Hasil tersebut adalah sebagai berikut:

Tabel 4.11 Pengecekan Kelayakan Jadwal Inisial Server Di Stasiun *Bottleneck* (Manual)

| Order | No. Order | Model | rib (jam ke-) | dib (jam ke-) | pj | STib | Urutan STib | Urutan distrb. | Server | Urutan ke-k | ekjs | Status | Flowtime |
|-------|-----------|----------------------------------|---------------|---------------|---------|-----------|-------------|----------------|--------|-------------|----------|--------|----------|
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8478 | 150.0113 | 12.2404 | 131.7709 | 1 | 1 | 1 | 1 | 18.2404 | Layak | 12.3996 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 150.0113 | 12.2404 | 131.7709 | 2 | 2 | 1 | 2 | 30.4809 | Layak | 24.6401 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 150.0113 | 12.2404 | 131.7709 | 3 | 3 | 1 | 3 | 42.7213 | Layak | 36.8805 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 5.8408 | 318.0113 | 12.2404 | 299.7709 | 4 | 4 | 1 | 4 | 54.9617 | Layak | 49.1210 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 2.0257 | 477.1871 | 3.6721 | 467.4650 | 5 | 5 | 1 | 5 | 58.6339 | Layak | 56.6082 |
| 8 | 1020609 | BK MATEMATIKA 38 POLOS (320) | 1.4807 | 478.8694 | 2.4841 | 470.4213 | 6 | 6 | 1 | 6 | 61.0820 | Layak | 59.6013 |
| 11 | 1020617 | KATHAN BESAR KECIL | 5.9089 | 676.1211 | 12.3934 | 657.7277 | 7 | 7 | 1 | 7 | 73.4754 | Layak | 67.5665 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 5.8408 | 696.0113 | 12.2404 | 687.7709 | 8 | 8 | 1 | 8 | 85.7158 | Layak | 79.8751 |
| 10 | 1020618 | LATHAN MEWARNA TANGGUNG | 6.5221 | 701.2144 | 13.7705 | 681.4440 | 9 | 9 | 1 | 9 | 99.4863 | Layak | 92.9643 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 0.7313 | 1712.2512 | 0.7650 | 705.4862 | 10 | 10 | 1 | 10 | 100.2514 | Layak | 99.5201 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 3.1157 | 1271.6726 | 6.1020 | 1259.5524 | 11 | 11 | 1 | 11 | 106.3716 | Layak | 103.2558 |

[illegible]

Gambar 4.42 Pengecekan Kelayakan Jadwal Inisial Server Di Stasiun *Bottleneck* (Program Aplikasi)

E. Penentuan Revisi Jadwal Inisial Server Di Stasiun *Bottleneck* dengan SPT

Pada tahap kelima dalam validasi program aplikasi penjadwalan produksi buku tulis pada jalur mesin 321 dengan menggunakan pendekatan *DBR* ini adalah dilakukan validasi proses penentuan revisi jadwal inisial server di stasiun *bottleneck* dengan *SPT*. Validasi proses penentuan revisi jadwal inisial server di stasiun *bottleneck* dengan *SPT* yaitu dengan membandingkan hasil antara perhitungan manual dan perhitungan dengan program aplikasi. Pada contoh ini, kondisinya layak sehingga tidak dilakukan revisi jadwal inisial server di SKB.

F. Penentuan Jadwal Produksi Buku Tulis

a) **Penjadwalan produksi buku tulis di Stasiun *Bottleneck***

Validasi program aplikasi penjadwalan produksi buku tulis di stasiun *bottleneck* ditentukan dari tahap-tahap sebelumnya, yaitu mulai validasi proses penentuan *mean lead time* order hingga validasi proses penentuan pengecekan kelayakan jadwal inisial dan proses revisi jadwal inisial server di stasiun *bottleneck* jika terdapat kondisi tidak layak. Hasil jadwal produksi buku tulis di stasiun *bottleneck* tersebut adalah jadwal produksi di SKB dengan kondisi layak ketika dilakukan pengecekan jadwal inisial atau jadwal produksi di SKB setelah revisi dengan *SPT* jika pada saat pengecekan jadwal inisial terdapat kondisi tidak layak.

b) Penjadwalan produksi di stasiun kerja *upstream* (*Backward Scheduling*)

Pada tahap selanjutnya dilakukan validasi program aplikasi penjadwalan produksi buku tulis. Validasi proses penjadwalan produksi buku tulis untuk stasiun kerja *non-bottleneck upstream* yaitu dilakukan validasi proses penentuan jadwal produksi buku tulis di stasiun 1 (Cetak Cover). Validasi ini dilakukan dengan membandingkan hasil jadwal produksi buku tulis yang dihitung manual dengan hasil dari program aplikasi. Hasil tersebut adalah sebagai berikut:

Tabel 4.12 Jadwal Produksi Buku Tulis Di Stasiun 1 (Cetak Cover)
(Manual)

| Jadwal Produksi di SKU Cetak Cover (j=1) dengan (mj=2) | | | | | | | |
|--|-----------|----------------------------------|-----------|----------|-------------|----------|---------|
| Order i | No. Order | Model | riks(j+1) | Server s | Urutan ke-k | ciksj | riksj |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 100.2514 | 1 | 9 | 100.2514 | 97.5263 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 99.4863 | 2 | 2 | 99.4863 | 99.1457 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 85.7158 | 1 | 8 | 85.7158 | 79.5845 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 73.4754 | 1 | 7 | 73.4754 | 68.0253 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 61.0820 | 1 | 6 | 61.0820 | 55.5637 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 58.6339 | 2 | 1 | 58.6339 | 57.5439 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 54.9617 | 1 | 5 | 54.9617 | 53.3267 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 42.7213 | 1 | 4 | 42.7213 | 37.2712 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 30.4809 | 1 | 3 | 30.4809 | 25.0308 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 18.2404 | 1 | 2 | 18.2404 | 12.7903 |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 6.0000 | 1 | 1 | 6.0000 | 0.5499 |

| JADWAL PRODUKSI DI STASIUN KERJA UPSTREAM (CETAK COVER) | | | | | | | | | |
|---|-----------|--------------------------------|-----------|--------|---------|----------|--------|--------|--|
| PERIODE ORDER | | MINGGU | 1 | BULAN | 04 | TAHUN | 2009 | | |
| i | No. Order | Model | Rksj (+1) | Pij | Riks | Ciksj | Server | Urutan | |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS | 6.0000 | 5.3900 | 0.6100 | 6.0000 | 1 | 1 | |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS | 18.1200 | 5.3900 | 12.7200 | 18.1200 | 1 | 2 | |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS | 30.2300 | 5.3900 | 24.8400 | 30.2300 | 1 | 3 | |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS | 42.3500 | 5.3900 | 36.9600 | 42.3500 | 1 | 4 | |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS | 54.4700 | 1.6200 | 52.8500 | 54.4700 | 1 | 5 | |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS | 58.1000 | 1.0800 | 57.0200 | 58.1000 | 1 | 6 | |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 60.5300 | 5.7400 | 54.7900 | 60.5300 | 2 | 1 | |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA | 73.4200 | 5.3900 | 68.0200 | 73.4200 | 1 | 7 | |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 85.5300 | 6.1300 | 79.4000 | 85.5300 | 1 | 8 | |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOM | 99.3000 | 0.3400 | 98.9600 | 99.3000 | 1 | 9 | |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA | 100.0600 | 2.7000 | 97.3600 | 100.0600 | 2 | 2 | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |

<
>

Proses
 Cetak
 Keluar

Gambar 4.43 Jadwal Produksi Buku Tulis Di Stasiun 1 (Cetak Cover)
(Program Aplikasi)

Selanjutnya adalah validasi proses penjadwalan untuk stasiun kerja *non-bottleneck upstream* yaitu validasi jadwal produksi buku tulis di stasiun 4 (Cetak Isi). Validasi proses penentuan jadwal produksi buku tulis di stasiun 4 (Cetak Isi)

dengan membandingkan hasil jadwal produksi buku tulis yang dihitung manual dengan hasil dari program aplikasi. Hasil tersebut adalah sebagai berikut:

Tabel 4.13 Jadwal Produksi Buku Tulis Di Stasiun 4 (Cetak Isi)
(Manual)

| Jadwal Produksi di SKU Cetak Isi (j=3) dengan (mj=6) | | | | | | | |
|--|-----------|----------------------------------|-----------|----------|-------------|----------|----------|
| Order i | No. Order | Model | riks(j+1) | Server s | Urutan ke-k | ciksj | riksj |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 105.2416 | 1 | 9 | 105.2416 | 101.4761 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 102.7137 | 2 | 2 | 102.7137 | 99.8172 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 99.8479 | 1 | 8 | 99.8479 | 99.1237 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 83.4559 | 2 | 1 | 83.4559 | 75.9248 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 76.3800 | 1 | 7 | 76.3800 | 73.7731 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 59.7907 | 1 | 6 | 59.7907 | 57.4735 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 56.6970 | 1 | 5 | 56.6970 | 53.2211 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 48.5055 | 1 | 4 | 48.5055 | 36.9193 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 36.2651 | 1 | 3 | 36.2651 | 24.6789 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 24.0246 | 1 | 2 | 24.0246 | 12.4384 |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 11.7842 | 1 | 1 | 11.7842 | 0.1980 |

[illegible]

Gambar 4.44 Jadwal Produksi Buku Tulis Di Stasiun 4 (Cetak Isi)
(Program Aplikasi)

Selanjutnya dilakukan validasi proses penjadwalan untuk stasiun kerja *non-bottleneck upstream* yaitu validasi jadwal produksi buku tulis di stasiun 6 (Lipat Isi). Validasi proses penentuan jadwal produksi buku tulis di stasiun 6 (Lipat Isi) dengan membandingkan hasil jadwal produksi buku tulis yang dihitung manual dengan hasil dari program aplikasi. Hasil tersebut adalah sebagai berikut:

Tabel 4.14 Jadwal Produksi Buku Tulis Di Stasiun 6 (Lipat Isi)
(Manual)

| Order i | No. Order | Model | riks(j+1) | Server s | Urutan ke-k | ciks _j | riks _j |
|---------|-----------|----------------------------------|-----------|----------|-------------|-------------------|-------------------|
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 109.1381 | 2 | 2 | 109.1381 | 105.2416 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 105.7110 | 1 | 8 | 105.7110 | 102.7137 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 100.5972 | 1 | 7 | 100.5972 | 99.8479 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 91.2489 | 1 | 6 | 91.2489 | 83.4559 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 79.0776 | 1 | 5 | 79.0776 | 76.3800 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 62.1886 | 3 | 1 | 62.1886 | 59.7907 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 60.4948 | 1 | 4 | 60.4948 | 48.5055 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 60.2938 | 2 | 1 | 60.2938 | 56.6970 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 48.2544 | 1 | 3 | 48.2544 | 36.2651 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 36.0139 | 1 | 2 | 36.0139 | 24.0246 |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 23.7735 | 1 | 1 | 23.7735 | 11.7842 |

[illegible]

Gambar 4.45 Jadwal Produksi Buku Tulis Di Stasiun 6 (Lipat Isi)
(Program Aplikasi)

c) Penjadwalan produksi di stasiun kerja *downstream* (*Forward Scheduling*)

Pada tahap selanjutnya dilakukan validasi program aplikasi penjadwalan produksi buku tulis untuk stasiun kerja *non-bottleneck downstream* yaitu dilakukan validasi proses penentuan jadwal produksi buku tulis di stasiun 3 (Embosst Cover). Validasi penentuan jadwal produksi buku tulis di stasiun 3 (Embosst Cover) dengan membandingkan hasil jadwal produksi buku tulis yang dihitung manual dengan hasil dari program aplikasi. Hasil tersebut adalah sebagai berikut:

Tabel 4.15 Jadwal Produksi Buku Tulis Di Stasiun 3 (Embosst Cover)
(Manual)

| Jadwal Produksi di SKDEmboss Cover (j=3) dengan (mj=3) | | | | | | | | |
|--|-----------|----------------------------------|-----------|--------|----------|-------------|----------|----------|
| Order i | No. Order | Model | ciks(j-1) | pij | Server s | Urutan ke-k | riksj | ciks=Rs |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 18.2404 | 4.6784 | 1 | 1 | 18.2404 | 22.9188 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 30.4809 | 4.6784 | 1 | 2 | 30.4809 | 35.1592 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 42.7213 | 4.6784 | 1 | 3 | 42.7213 | 47.3997 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 54.9617 | 4.6784 | 1 | 4 | 54.9617 | 59.6401 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 58.6339 | 1.4035 | 2 | 1 | 58.6339 | 60.0374 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 61.0820 | 0.9357 | 1 | 5 | 61.0820 | 62.0176 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 73.4754 | 4.7368 | 1 | 6 | 73.4754 | 78.2123 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 85.7158 | 4.6784 | 1 | 7 | 85.7158 | 90.3942 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 99.4863 | 5.2632 | 1 | 8 | 99.4863 | 104.7495 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 100.2514 | 0.2924 | 2 | 2 | 100.2514 | 100.5438 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 106.3716 | 2.3392 | 1 | 9 | 106.3716 | 108.7108 |

JADWAL PRODUKSI DI STASIUN KERJA DOWNSTREAM (EMBOSS COVER)

PERIODE ORDER
MINGGU 1
BULAN 04
TAHUN 2009

| i | No. Order | Model | Ciks(j-1) | Pij | Riks | Ciks | Server | Urutan |
|----|-----------|---------------------------------|-----------|--------|----------|----------|--------|--------|
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS | 18.1200 | 4.6300 | 18.1200 | 22.7500 | 1 | 1 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS | 30.2300 | 4.6300 | 30.2300 | 34.8700 | 1 | 2 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS | 42.3500 | 4.6300 | 42.3500 | 46.9800 | 1 | 3 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS | 54.4700 | 4.6300 | 54.4700 | 59.1000 | 1 | 4 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS | 58.1000 | 1.3900 | 58.1000 | 59.4900 | 2 | 1 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS | 60.5300 | 0.9300 | 60.5300 | 61.4500 | 1 | 5 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 73.4200 | 4.9300 | 73.4200 | 78.3400 | 1 | 6 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA | 85.5300 | 4.6300 | 85.5300 | 90.1700 | 1 | 7 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 99.3000 | 5.2600 | 99.3000 | 104.5700 | 1 | 8 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 100.0600 | 0.2900 | 100.0600 | 100.3500 | 2 | 2 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA | 106.1200 | 2.3200 | 106.1200 | 108.4400 | 1 | 9 |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |

Proses
Cetak
Keluar

Gambar 4.46 Jadwal Produksi Buku Tulis Di Stasiun 3 (Embosst Cover)
(Program Aplikasi)

Pada tahap selanjutnya dilakukan validasi proses penentuan jadwal produksi buku tulis di stasiun 5 (Potong Cover). Validasi proses penentuan jadwal produksi buku tulis di stasiun 5 (Potong Cover) dengan membandingkan hasil jadwal produksi buku tulis yang dihitung manual dengan hasil dari program aplikasi. Hasil tersebut adalah sebagai berikut:

Tabel 4.16 Jadwal Produksi Buku Tulis Di Stasiun 5 (Potong Cover)
(Manual)

| Jadwal Produksi di SKD Potong Cover (j=5) dengan (mj=2) | | | | | | | | |
|---|-----------|----------------------------------|-----------|--------|----------|-------------|----------|----------|
| Order i | No. Order | Model | ciks(j-1) | pij | Server s | Urutan ke-k | riksj | ciks=Rs |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 22.9188 | 0.8547 | 1 | 1 | 22.9188 | 23.7735 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 35.1592 | 0.8547 | 1 | 2 | 35.1592 | 36.0139 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 47.3997 | 0.8547 | 1 | 3 | 47.3997 | 48.2544 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 59.6401 | 0.8547 | 1 | 4 | 59.6401 | 60.4948 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 60.0374 | 0.2564 | 2 | 1 | 60.0374 | 60.2938 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 62.0176 | 0.1709 | 1 | 5 | 62.0176 | 62.1886 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 78.2123 | 0.8654 | 1 | 6 | 78.2123 | 79.0776 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 90.3942 | 0.8547 | 1 | 7 | 90.3942 | 91.2489 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 104.7495 | 0.9615 | 1 | 8 | 104.7495 | 105.7110 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 100.5438 | 0.0534 | 2 | 2 | 100.5438 | 100.5972 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 108.7108 | 0.4274 | 1 | 9 | 108.7108 | 109.1381 |

[illegible]

Gambar 4.47 Jadwal Produksi Buku Tulis Di Stasiun 5 (Potong Cover)
(Program Aplikasi)

Pada tahap selanjutnya dilakukan validasi proses penentuan jadwal produksi buku tulis di stasiun 7 (Mesin 321). Validasi proses penentuan jadwal produksi buku tulis di stasiun 7 (Mesin 321) dengan membandingkan hasil jadwal produksi buku tulis yang dihitung manual dengan hasil dari program aplikasi. Hasil tersebut adalah sebagai berikut:

Tabel 4.17 Jadwal Produksi Buku Tulis Di Stasiun 7 (Mesin 321)
(Manual)

| Order i | No. Order | Model | ciks(j-1) | pij | Server s | Urutan ke-k | riksj | ciks=j-Rsj |
|---------|-----------|----------------------------------|-----------|---------|----------|-------------|----------|------------|
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 23.7735 | 11.7895 | 1 | 1 | 23.7735 | 35.5630 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 36.0139 | 11.7895 | 1 | 2 | 36.0139 | 47.8034 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 48.2544 | 11.7895 | 1 | 3 | 48.2544 | 60.0438 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 60.4948 | 11.7895 | 1 | 4 | 60.4948 | 72.2843 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 60.2938 | 3.5368 | 2 | 1 | 60.2938 | 63.8306 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 62.1886 | 2.3579 | 3 | 1 | 62.1886 | 64.5465 |
| 11 | 1020617 | LATHAN MEWARNAI KECIL | 79.0776 | 10.6105 | 1 | 5 | 79.0776 | 89.6882 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 91.2489 | 11.7895 | 1 | 6 | 91.2489 | 103.0384 |
| 10 | 1020618 | LATHAN MEWARNA TANGGUNG | 105.7110 | 5.8947 | 1 | 7 | 105.7110 | 111.6058 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 100.5972 | 0.7368 | 2 | 2 | 100.5972 | 101.3340 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 109.1381 | 5.8947 | 2 | 3 | 109.1381 | 115.0329 |

[illegible]

Gambar 4.48 Jadwal Produksi Buku Tulis Di Stasiun 7 (Mesin 321)
(Program Aplikasi)

G. Penentuan Performansi Jadwal Produksi Buku Tulis

Pada tahap terakhir dilakukan validasi penentuan performansi jadwal produksi buku tulis di jalur mesin 321. Validasi penentuan performansi jadwal produksi buku tulis di jalur mesin 321 dengan membandingkan hasil jadwal produksi buku tulis yang dihitung manual dengan hasil dari program aplikasi. Hasil tersebut adalah sebagai berikut:

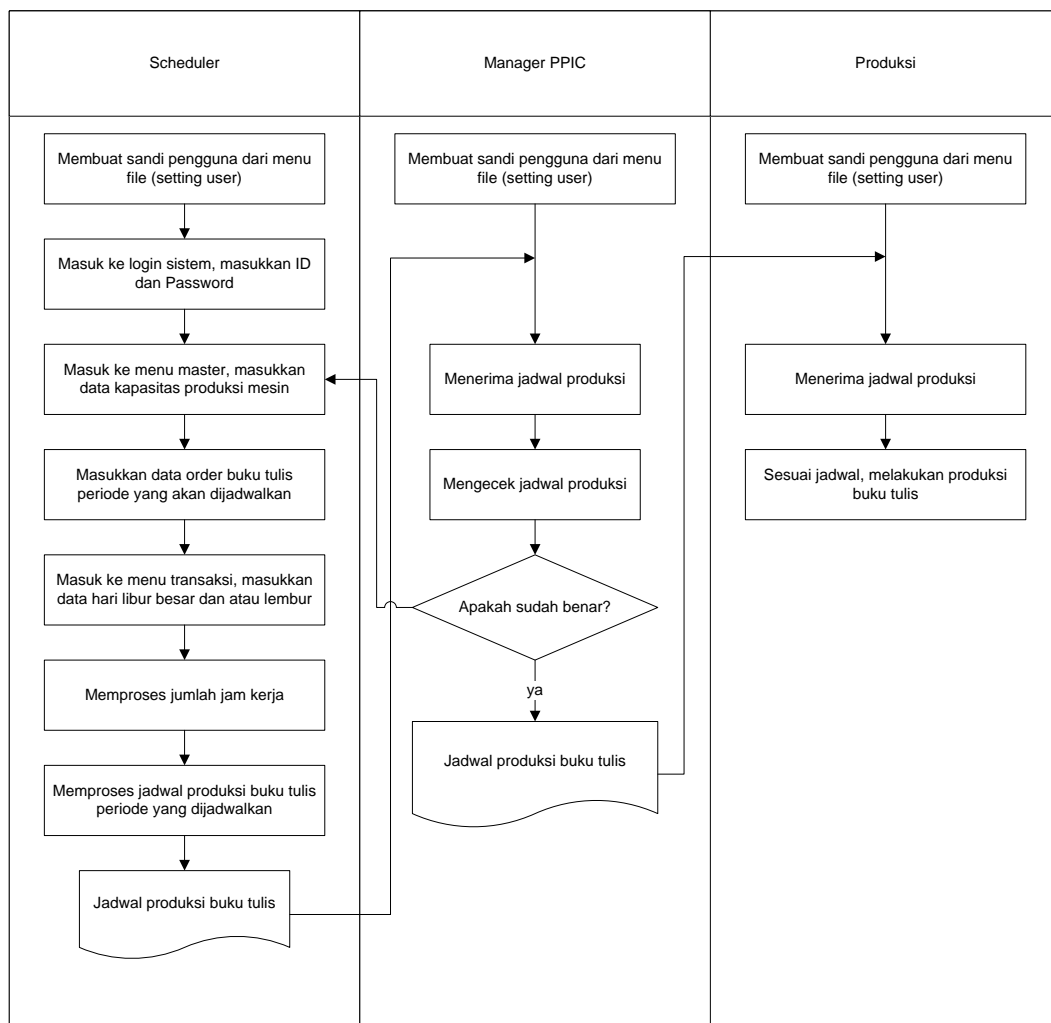
Tabel 4.18 Performansi Jadwal Produksi Buku Tulis (Manual)

| Order i | No. Order | Model | ciksj | | riksj | | Flowtime | |
|---------|-----------|----------------------------------|----------|------------|---------|------------|----------|------|
| | | | jam | tanggal | jam | tanggal | jam | hari |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 35.5630 | 14-04-2009 | 0.1980 | 13-04-2009 | 35.3650 | 2 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 47.8034 | 14-04-2009 | 12.4384 | 13-04-2009 | 35.3650 | 2 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 60.0438 | 15-04-2009 | 24.6789 | 14-04-2009 | 35.3650 | 2 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 72.2843 | 15-04-2009 | 36.9193 | 14-04-2009 | 35.3650 | 2 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 101.3340 | 17-04-2009 | 99.1237 | 17-04-2009 | 2.2103 | 1 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 115.0329 | 17-04-2009 | 97.5263 | 17-04-2009 | 17.5065 | 1 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 103.0384 | 17-04-2009 | 68.0253 | 15-04-2009 | 35.0131 | 2 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 64.5465 | 15-04-2009 | 57.4735 | 15-04-2009 | 7.0730 | 1 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 63.8306 | 15-04-2009 | 53.2211 | 15-04-2009 | 10.6095 | 1 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 111.6058 | 17-04-2009 | 79.5845 | 16-04-2009 | 32.0213 | 2 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 89.6882 | 16-04-2009 | 55.5637 | 15-04-2009 | 34.1244 | 2 |

Gambar 4.49 Performansi Jadwal Produksi Buku Tulis
(Program Aplikasi)

4.6 Prosedur Pengoperasian Program Aplikasi Penjadwalan Produksi

Program aplikasi penjadwalan produksi buku tulis pada jalur 321 dibuat sesuai algoritma penjadwalan produksi buku tulis dengan menggunakan pendekatan *DBR*. Program aplikasi yang dibuat ini dibuat untuk penjadwalan produksi per periode (periode mingguan). Adapun prosedur pengoperasian program aplikasi penjadwalan produksi buku tulis yaitu seperti pada gambar 4.50 dibawah ini.



Gambar 4.50 Prosedur Pengoperasian Program Aplikasi Penjadwalan Buku Tulis

Prosedur pengoperasian program aplikasi penjadwalan buku tulis ini diberlakukan pada pembuat jadwal (*scheduler*), manager PPIC dan bagian produksi. Sebelum menggunakan atau mengakses program aplikasi ini, harus dibuat ID dan sandi pengguna untuk menjaga kerahasiaan dokumen yang dibuat. Selanjutnya *scheduler* memasukkan data kapasitas produksi mesin-mesin pada jalur mesin 321. Setelah selesai, lalu memasukkan data order buku tulis. Data order buku tulis tersebut yaitu: kode buku, nama buku, periode order, tanggal saat masuk order, tanggal periode awal order dijadwalkan di lini produksi, tanggal target selesai order, jumlah order, jumlah isi buku tulis (lembar) dan jumlah cover buku tulis (lembar). Setelah selesai mengisi data order buku tulis, selanjutnya memasukkan data hari libur besar dan atau lembur dan memproses jam kerja.

Setelah selesai memasukkan data-data, selanjutnya dapat dilakukan proses penjadwalan buku tulis pada masing-masing stasiun kerja dengan menekan tombol “proses”, selanjutnya secara otomatis data jadwal produksi buku tulis masing-masing stasiun kerja jadi.

Setelah jadwal produksi jadi, selanjutnya jadwal dicek oleh manager PPIC. Apabila jadwal produksi tersebut sudah benar maka dapat langsung diteruskan ke bagian produksi untuk dilakukan proses produksi. Apabila terjadi kesalahan jadwal maka dilakukan penjadwalan ulang dengan mengecek ulang data kapasitas produksi mesin hingga memproses jam kerja.



BAB V

ANALISIS DAN INTEPRETASI HASIL

5.1 Analisis Sistem Penjadwalan Produksi Buku tulis pada jalur mesin 321

Sistem produksi buku tulis pada jalur mesin 321 di PT. Solo Murni adalah *mixed model repetitive flowshop*, artinya bahwa lini produksi tersebut mengerjakan berbagai jenis model dalam satu lini produksi. Jenis model tersebut yaitu Buku Tulis Kiki, Buku Diktat dan Buku Boxy. Sedangkan sistem produksi *flowshop* ditunjukkan dengan pengerjaan order dari stasiun pertama hingga stasiun terakhir dalam proses produksinya dengan tidak mengalami proses balik ke bagian hulu atau bagian produksi sebelumnya dan setiap stasiun kerja memiliki jumlah *server* (mesin) yang berbeda-beda. Stasiun kerja tersebut berturut-urut adalah stasiun kerja Cetak Cover, Vurnish Cover, Embosst Cover, Potong Cover, Cetak Isi, Lipat Isi dan Mesin 321.

Pada lini produksi buku tulis di jalur mesin 321 terdapat sistem *bottleneck* yaitu pada stasiun kerja Vurnish Cover. Hal ini terjadi karena pada stasiun kerja sebelumnya (Stasiun Kerja Cetak Cover) memiliki kapasitas produksi yang lebih besar, yaitu sebesar 68.500 lembar plano/hari, kemudian ditumpuk di SKB yang hanya memiliki kapasitas produksi sebesar 30.500 lembar plano/hari. Penumpukan tersebut yang menjadikan stasiun kerja Vurnish Cover menjadi *bottleneck* (SKB) yaitu sebesar 38.000 lembar plano/hari.

Adapun metode penjadwalan yang digunakan perusahaan sekarang ini adalah dengan *forward scheduling*. Metode *forward scheduling* yaitu penjadwalan produksi dilakukan mulai dari stasiun kerja awal berturut-turut hingga stasiun kerja akhir. Pada metode penjadwalan tersebut, order-order yang datang dikumpulkan dahulu baru kemudian di jadwalkan oleh Departemen PPIC per periode. Pengumpulan order-order yang datang dilakukan pada hari Jumat minggu sebelum sampai dengan Sabtu minggu sesudah. Setelah order terkumpul kemudian baru dijadwalkan untuk proses produksi pada hari Senin sampai dengan Sabtu. Selain itu, order-order yang datang diproduksi dengan urutan atau aturan FCFS (*First Come First Serve*), yaitu order yang pertama datang maka order juga pertama diproduksi dahulu.

Metode *forward scheduling* yang digunakan perusahaan tidak menjamin pengerjaan order yang dapat meminimasi *mean flowtime* pada SKB, apalagi di SKB mengalami *bottleneck*. Hal ini karena penjadwalan produksi diawali dari stasiun kerja 1, yaitu Stasiun Kerja Cetak Cover, yang tidak berstatus *bottleneck*. Penjadwalan produksi dengan pendekatan ini menimbulkan kedatangan order di SKB memiliki urutan yang acak yang tidak menjamin minimasi *mean flowtime* karena penjadwalan tidak direncanakan dari SKB.

Sedangkan pada rancangan ulang pada sistem penjadwalan ini menggunakan pendekatan DBR yang pengaturan sistem produksinya dengan mengkonsentrasikan pengendalian di SKB. Jadi SKB digunakan sebagai penggerak dalam penjadwalan produksi buku tulis pada sistem secara menyeluruh. Hal tersebut dapat dikatakan bahwa model penjadwalan produksinya berfisat campuran yaitu antara *forward scheduling* dan *backward scheduling*. Jadi penjadwalan produksi dimulai dari SKB yaitu Vurnish Cover. Selanjutnya untuk SK *Upstream* menggunakan *backward scheduling* dan untuk SK *Downstream* menggunakan *forward scheduling*.

Pendekatan DBR ini mampu mengurangi rata-rata *flowtime* (*mean flowtime*) order dalam proses baik pada SKB maupun sistem secara keseluruhan. Pengurangan *mean flowtime* bisa dilakukan karena minimasi *mean flowtime* dimaksimalkan pada SKB. Pengurangan *mean flowtime* pada SKB berarti juga akan mengurangi *mean flowtime* pada sistem secara keseluruhan karena titik kontrol dalam sistem *bottleneck* adalah SKB (stasiun kerja Vurnish Cover). Rata-rata *flowtime* (*mean flowtime*) order sistem keseluruhan bisa dikurangi dengan menerapkan pendekatan ini. Hal ini ditunjukkan dari hasil simulasi penjadwalan produksi buku tulis dengan menggunakan pendekatan DBR yang telah dilakukan.

Penggunaan metode SPT (*Shortest Processing Time*) dan penerapan aturan prioritas berorientasi *due date* (target), yaitu *slack time*, pada penjadwalan produksi SKB sangat berarti untuk meminimasi *mean flowtime*. *Slack time* digunakan sebagai aturan prioritas berorientasi target karena selain mempertimbangkan target, *slack time* juga mempertimbangkan waktu proses order, $ST = due\ date - present\ date - waktu\ proses$. Waktu proses perlu

dipertimbangkan dalam pengurutan prioritas pengerjaan buku tulis karena waktu proses masing masing order yang berbeda.

Selain itu dengan pendekatan *DBR* ini nantinya mampu mengurangi *WIP* dilantai produksi. Pengurangan *WIP* dilantai produksi dapat dicapai karena penjadwalan dikonsentrasikan pada SKB (stasiun kerja Vurnish Cover) sebagai titik kontrol. Berdasarkan jadwal produksi pada SKB, order-order dijadwalkan dengan pendekatan *backward scheduling* menuju ke stasiun kerja *upstream*. Penjadwalan dengan pendekatan *backward scheduling* akan meminimasi *WIP* dilantai produksi karena order masuk ke lantai produksi disesuaikan dengan kapasitas produksi pada SKB sehingga order datang sesaat sebelum dikerjakan di SKB.

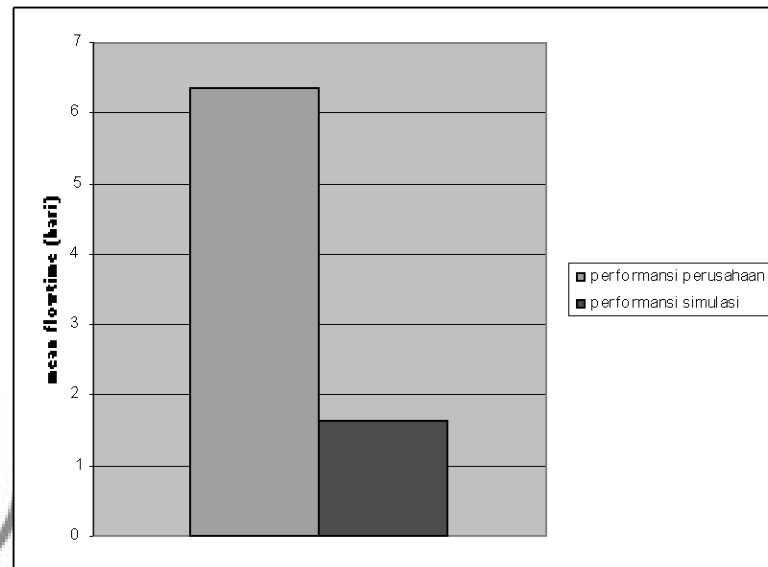
Penerapan aturan prioritas pengurutan berorientasi pada waktu tinggal atau *flowtime*, yaitu *shortest processing time (SPT)* untuk meminimasi *mean flowtime*. Kelebihan penerapan penjadwalan produksi buku tulis pada jalur mesin 321 dengan menggunakan pendekatan *DBR* ditunjukkan dengan hasil simulasi dimana *mean flowtime* bisa dikurangi. Ini berarti mengindikasikan bahwa jumlah *WIP* juga bisa dikurangi.

Akan tetapi sistem penjadwalan produksi buku tulis dengan menggunakan pendekatan *DBR* ini adalah algoritma yang dikembangkan dengan pendekatan heuristik sehingga hasil yang diperoleh tidak selalu optimal.

5.2 Analisis Performansi Pada Lini Produksi Buku tulis pada jalur mesin 321

Berdasarkan data perusahaan diketahui bahwa di stasiun kerja Vurnish Cover memiliki kapasitas produksi yang rendah yaitu sebesar 30.500 lembar plano/hari. Apalagi metode penjadwalan yang digunakan perusahaan adalah *forward scheduling*. Dengan hal tersebut menyebabkan penumpukan *WIP* di SKB karena stasiun kerja sebelum SKB (Stasiun Kerja Cetak Cover) memiliki kapasitas produksi yang lebih besar yaitu sebesar 68.500 lembar plano/hari. Hal tersebut menyebabkan lamanya waktu tinggal atau *flowtime* order di lantai produksi. Rata-rata waktu tinggal (*mean flowtime*) order di lantai produksi dari order-order yang masuk pada periode I tanggal 8-10 April 2009 adalah 6 hari dengan rentang waktu tinggal atau *flowtime* order di lantai produksi bervariasi dari

2 hari sampai 11 hari. Simulasi dengan pendekatan *DBR* dalam penjadwalan produksi buku tulis pada jalur mesin 321 menunjukkan *mean flowtime* order di lantai produksi yang lebih baik, yaitu sebesar 2 hari dengan rentang *flowtime* order di lantai produksi bervariasi dari 1 hari sampai 2 hari. Perbandingan *flowtime* order di lantai produksi antara data perusahaan dan hasil simulasi ditunjukkan pada Gambar 5.1. dan Tabel 5.1.



Gambar 5.1 Grafik Perbandingan *Mean Flowtime*

Tabel 5.1 Perbandingan Waktu Tinggal (*Flowtime*) di Lantai Produksi

| Order | No. Order | Model | Flowtime Simulasi | Flowtime perusahaan |
|-------------------------|-----------|----------------------------------|-------------------|---------------------|
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 2 | 3 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 2 | 2 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 2 | 2 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 2 | 10 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 1 | 8 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 1 | 11 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 2 | 9 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 1 | 6 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 1 | 7 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 2 | 5 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 2 | 7 |
| rata-rata (hari) | | | 2 | 6 |

Rata-rata waktu tinggal atau *mean flowtime* order di lantai produksi bisa dikurangi hingga mencapai 62.86 % atau menghemat rata-rata waktu tinggal atau *mean flowtime* order di lantai produksi sebesar 4 hari. Perbaikan rata-rata waktu tinggal atau *mean flowtime* order di lantai produksi ini akan meningkatkan performansi PT. Solo Murni untuk jumlah *WIP* di lantai produksi. Hal ini karena

semakin kecil *mean flowtime* order di lantai produksi berarti semakin pendek atau singkat waktu yang diperlukan order dalam proses produksi di lantai produksi. Kondisi ini mengindikasikan bahwa penumpukan material atau order di lantai produksi bisa dikurangi sehingga jumlah *WIP* di lantai produksi juga bisa dikurangi. Selain itu, pemborosan material bisa dikurangi dan penambahan ruang untuk penempatan *WIP* atau pengerjaan order tidak perlu dilakukan.

Tabel 5.2 Perbandingan Keterlambatan (*Lateness*) di Lantai Produksi

| Order i | No. Order | Model | Target Selesai | Selesai Perusahaan | Selesai Simulasi | Lateness (hari) | |
|---------|-----------|----------------------------------|----------------|--------------------|------------------|-----------------|----------|
| | | | | | | Perusahaan | Simulasi |
| 1 | 1020596 | BK TULIS KIKY ISI 38 POLOS (320) | 21-04-2009 | 17-04-2009 | 14-04-2009 | 0 | 0 |
| 2 | 1020597 | BK TULIS KIKY ISI 38 POLOS (320) | 21-04-2009 | 17-04-2009 | 14-04-2009 | 0 | 0 |
| 3 | 1020599 | BK TULIS KIKY ISI 38 POLOS (320) | 21-04-2009 | 20-04-2009 | 15-04-2009 | 0 | 0 |
| 4 | 1020600 | BK TULIS KIKY ISI 38 POLOS (320) | 30-04-2009 | 28-04-2009 | 15-04-2009 | 0 | 0 |
| 5 | 7100915 | BK TLS 38 X-FILE YAYASAN SUTOMO | 23-05-2009 | 28-05-2009 | 17-04-2009 | 5 | 0 |
| 6 | 1020606 | BK TULIS KIKY 26 WARNA (400) | 24-06-2009 | 22-06-2009 | 17-04-2009 | 0 | 0 |
| 7 | 1020607 | BK TULIS KIKY 26 WARNA (400) | 24-05-2009 | 22-05-2009 | 17-04-2009 | 0 | 0 |
| 8 | 1020609 | BK KOTAK BESAR 38 POLOS (320) | 10/5/2009 | 4/5/2009 | 15-04-2009 | 0 | 0 |
| 9 | 1020611 | BK MATEMATIKA 38 POLOS (320) | 10/5/2009 | 2/5/2009 | 15-04-2009 | 0 | 0 |
| 10 | 1020618 | LATIHAN MEWARNA TANGGUNG | 23-05-2009 | 21-05-2009 | 17-04-2009 | 0 | 0 |
| 11 | 1020617 | LATIHAN MEWARNAI KECIL | 22-05-2009 | 21-05-2009 | 16-04-2009 | 0 | 0 |

Berdasarkan tabel 5.2 tersebut diatas, bahwa terdapat keterlambatan (*lateness*) penyelesaian order yang terjadi di perusahaan. Keterlambatan tersebut terjadi pada penyelesaian order ke 5, dengan keterlambatan sebesar 5 hari. Sedangkan pada hasil simulasi dengan menggunakan pendekatan DBR tidak terdapat keterlambatan penyelesaian order. Jadi penjadwalan produksi buku tulis pada jalur mesin 321 dengan menggunakan pendekatan DBR tidak hanya meminimasi *mean flowtime* melainkan juga meminimasi keterlambatan penyelesaian order yang terjadi di PT. Solo Murni.

Perbaikan *mean flowtime* order maupun *lateness* penyelesaian order di lantai produksi bisa dicapai karena pada dasarnya penerapan pendekatan DBR pada penjadwalan produksi buku tulis pada jalur mesin 321 akan mampu mengendalikan jumlah order yang masuk ke lini produksi. Pengendalian ini bisa dilakukan karena penjadwalan dikonsentrasikan pada SKB yaitu Stasiun Kerja Vurnish Cover. Berdasarkan jadwal produksi pada SKB, order-order dijadwalkan dengan pendekatan *backward scheduling* menuju ke stasiun kerja *upstream* yaitu stasiun kerja Cetak Cover. Penjadwalan dengan pendekatan *backward scheduling*, akan meminimasi *WIP* dilantai produksi karena pada penjadwalan DBR order masuk ke lantai produksi disesuaikan dengan kapasitas produksi pada SKB sehingga order datang sesaat sebelum dikerjakan di SKB.

Mekanisme penyampaian perintah memasukkan order ke lini produksi dalam algoritma penjadwalan *DBR* disebut *rope*. Mekanisme ini sangat penting untuk mengendalikan jumlah *WIP* di rantai produksi. Keberadaan mekanisme *rope* ini akan menjaga agar order yang masuk ke lini produksi disesuaikan dengan kapasitas SKB. Dengan demikian SKB bisa bekerja dalam kapasitas penuh. Mekanisme *rope* dilakukan berdasarkan mekanisme *buffer* yang disediakan antara SKB dan stasiun kerja 1 (paling awal) yaitu stasiun kerja Cetak Cover.

Buffer dalam penjadwalan produksi buku tulis pada jalur mesin 321 dengan menggunakan pendekatan *DBR* merupakan selang waktu antara saat *release* di stasiun kerja Cetak Cover dengan SKB. Dalam penelitian ini *buffer* yang digunakan adalah *time buffer*. Penggunaan *time buffer* mampu mengakomodasi berbagai order buku tulis yang memiliki variasi waktu proses yang cukup tinggi. Selain itu keberadaan *buffer* berupa waktu mampu memprediksikan saat *release* order ke lini produksi sehingga *WIP* bisa dikendalikan.

Perhitungan *time buffer* dilakukan dengan pendekatan Zijm-Buitenhek. Algoritma Zijm-Buitenhek merupakan algoritma untuk memperkirakan *mean lead time* yang mencakup *waiting time* dan waktu proses order berdasarkan teori antrian. Penggunaan algoritma Zijm-Buitenhek sangat membantu dalam perhitungan *earliest bottleneck release* ($r_i^{(b)}$) dan *bottleneck due date* ($d_i^{(b)}$) yang digunakan sebagai dasar penjadwalan produksi pada SKB.

Minimasi *mean flowtime* order di rantai produksi yang dicapai dengan penerapan pendekatan *DBR* pada penjadwalan produksi buku tulis pada jalur mesin 321 bisa ditingkatkan dengan menerapkan aturan prioritas pengurutan berorientasi pada waktu tinggal atau *flowtime*. Aturan prioritas pengurutan tersebut adalah *shortest processing time* (*SPT*). Kombinasi antara penerapan pendekatan *DBR* dan aturan prioritas pengurutan berorientasi pada *flowtime* tersebut mampu memperbaiki rata-rata waktu tinggal atau *mean flowtime* di rantai produksi.

5.3 Analisis Program Aplikasi Penjadwalan Produksi Buku tulis pada jalur mesin 321

Pada perancangan program aplikasi penjadwalan produksi buku tulis ini dilakukan pembuatan konstruksi program. Konstruksi program merupakan

penyusun program aplikasi penjadwalan produksi yang dibuat. Pada konstruksi program disusun dari langkah-langkah logis hingga dapat memecahkan masalah. Perancangan program aplikasi penjadwalan produksi buku tulis yang dibuat, yang meliputi diagram alir program, *syntax* program dan *interface* program yang telah dirancang.

Program aplikasi tersebut dibuat sesuai algoritma penjadwalan produksi buku tulis pada jalur mesin 321 dengan menggunakan pendekatan *DBR*. Program aplikasi yang dibuat ini telah dirancang untuk penjadwalan produksi per periode (periode mingguan). Pada halaman utama program aplikasi terlihat menu utama yang terdiri dari file, master, transaksi dan bantuan.

Pada menu file terdapat menu *setting user* yaitu untuk mengatur (*setting*) pengguna program berikut dengan sandi pengguna; *log in* yaitu menu yang disediakan untuk dapat mengakses program aplikasi setelah dilakukan *setting user* sebelumnya; *log out* yaitu menu yang disediakan untuk keluar dari akses program aplikasi ini; dan *exit* yaitu menu yang disediakan untuk keluar dari program aplikasi penjadwalan buku tulis yang telah dibuat.

Pada menu *master* terdapat menu *input* data kapasitas produksi yaitu merupakan file *master* masing-masing kapasitas produksi untuk jalur mesin 321 dan jumlah *server* (mesin) masing-masing stasiun kerja dan *input* data buku yaitu file *master* order buku tulis yang masuk untuk diproduksi di lantai produksi.

Pada menu transaksi ini merupakan menu yang menampilkan proses penjadwalan buku tulis dengan menggunakan pendekatan *DBR*. Menu transaksi ini terdiri dari menu *input* data order produksi yaitu menu yang memasukkan identitas order buku tulis (nomer order, model, tanggal order, tanggal produksi, tanggal target, jumlah order, jumlah isi buku tulis dan jumlah cover buku tulis); *input* data hari libur yaitu menu yang memasukkan hari libur nasional karena ini terkait dengan jam kerja; *routing* dan waktu proses yaitu menu yang menampilkan hasil perhitungan waktu proses produksi order buku tulis di masing-masing stasiun kerja; *arrival rate* yaitu menu yang menampilkan hasil perhitungan *arrival rate* order di masing-masing stasiun kerja; *average workload* yaitu menu yang menampilkan hasil perhitungan *workload* order di masing-masing stasiun kerja berikut dengan rekap nilai *average workload* stasiun kerja; *mean waiting time*

yaitu menu yang menampilkan hasil perhitungan *waiting time* order di masing-masing stasiun kerja; *mean lead time* yaitu menu yang menampilkan hasil perhitungan *lead time* order di masing-masing stasiun kerja; stasiun kerja *bottleneck* yaitu menu yang menampilkan hasil *bottleneck release* order dan *bottleneck due date* order; distribusi order yaitu menu yang menampilkan pendistribusian order di stasiun kerja *bottleneck*; jadwal produksi di SKB yaitu menu yang menampilkan pengecekan kelayakan jadwal produksi di SKB (jadwal inisial); jadwal inisial sebelum atau sesudah revisi yaitu menu yang menampilkan hasil jadwal produksi di SKB setelah dicek kelayakannya, jika layak maka jadwal produksi yang digunakan adalah jadwal sebelum revisi dan jika tidak layak maka jadwal yang digunakan adalah jadwal setelah revisi; jadwal produksi di SKU – Cetak Cover yaitu menu yang menampilkan hasil jadwal produksi di stasiun kerja Cetak Cover; jadwal produksi di SKU – Lipat Isi yaitu menu yang menampilkan hasil jadwal produksi di stasiun kerja Lipat Isi; jadwal produksi di SKU – Cetak Isi yaitu menu yang menampilkan hasil jadwal produksi di stasiun kerja Cetak Isi; jadwal produksi di SKD – Embosst Cover yaitu menu yang menampilkan hasil jadwal produksi di stasiun kerja Embosst Cover; jadwal produksi di SKD – Potong Cover yaitu menu yang menampilkan hasil jadwal produksi di stasiun kerja Potong Cover; jadwal produksi di SKD – Mesin 321 yaitu menu yang menampilkan hasil jadwal produksi di stasiun kerja Mesin 321 dan menu performansi order produksi yaitu menu yang menampilkan hasil performansi jadwal produksi pada sistem keseluruhan.

Pada program aplikasi ini dalam validasinya dilakukan perbandingan perhitungan antara perhitungan manual yang dihitung sesuai algoritma penjadwalan dan perhitungan dengan program aplikasi. Hasil perhitungannya secara umum adalah sama. Yang membedakan adalah nilai angka dibelakang koma. Hal ini disebabkan karena jika pada perhitungan manual yang dilakukan di Excel ini, pada proses perhitungannya nilai angka dibelakang koma diproses semua. Sedangkan pada program aplikasi, pada proses perhitungannya hanya nilai angka yang tertera.

5.4 Interpretasi Hasil

Pada penelitian ini dapat dilakukan interpretasi hasil penelitian dari rancang ulang sistem penjadwalan produksi buku tulis pada jalur mesin 321 di PT. Solo Murni dengan menggunakan pendekatan *DBR* dan program aplikasi yang telah dibuat. Interpretasi tersebut bermaksud untuk memperkuat pengertian terhadap penafsiran hasil penelitian tersebut.

Pada program aplikasi penjadwalan produksi ini, menunjukkan *workload* order di masing-masing stasiun kerja dalam produksi buku tulis pada jalur mesin 321. Adanya informasi nilai *workload* order di masing-masing stasiun kerja pada program aplikasi dapat mempermudah dalam pengontrolan rantai produksi. *Workload* order di stasiun kerja yang nilainya lebih besar sama dengan 1 ($\rho_j \geq 1$) artinya bahwa pada stasiun kerja tersebut sudah tidak dapat menampung pengerjaan order, sehingga sisa order dikerjakan pada periode berikutnya. Jadi dengan adanya informasi nilai *workload* order maka digunakan sebagai pendeteksi nilai beban kerja stasiun.

Program aplikasi penjadwalan produksi ini menunjukkan pengurutan pengerjaan order di masing-masing stasiun kerja dalam produksi buku tulis pada jalur mesin 321. Adanya informasi pengurutan pengerjaan order di masing-masing stasiun kerja pada program aplikasi maka dapat mempermudah dalam pengontrolan rantai produksi. Pengurutan pengerjaan order di masing-masing stasiun kerja artinya bahwa order dikerjakan pada stasiun kerja tersebut sudah sesuai dengan urutan sehingga tidak akan terjadi tabrakan dalam pengerjaan order. Jadi dengan adanya informasi pengurutan pengerjaan order di masing-masing stasiun kerja pada program aplikasi maka order-order tersebut didistribusikan merata sesuai alokasinya dan sehingga tidak bertabrakan antar order yang didistribusikan.

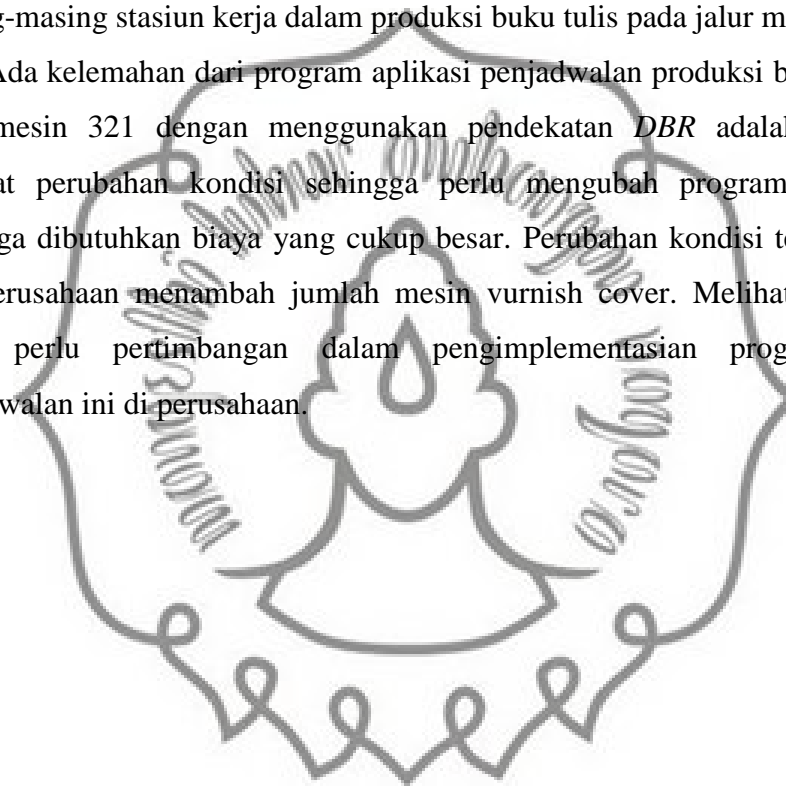
5.5 Rencana Implementasi

Pada hasil penelitian ini merancang ulang sistem penjadwalan produksi buku tulis pada jalur mesin 321 di PT. Solo Murni dengan menggunakan pendekatan *DBR* serta membuat program aplikasinya berdasarkan rancangan ulang tersebut, memungkinkan suatu rencana implementasi di perusahaan. Implementasi tersebut

dilakukan di Departemen PPIC PT. Solo Murni. Pada pengimplementasiannya perlu dipersiapkan sumber daya manusia atau karyawan yaitu dengan memberikan pelatihan terhadap karyawan tentang program aplikasi yang dibuat. Selain itu juga dipersiapkan pemasangan dan instalasi program aplikasi.

Pengimplementasian rancangan ulang tersebut memiliki pengaruh terhadap lama pengerjaan order di rantai produksi, waktu pembuatan jadwal produksi buku tulis yang nantinya akan diedarkan di rantai produksi, akses informasi jadwal produksi yang telah dibuat. Hal ini didukung dengan adanya informasi nilai *workload* order di masing-masing stasiun kerja dan urutan pengerjaan order di masing-masing stasiun kerja dalam produksi buku tulis pada jalur mesin 321.

Ada kelemahan dari program aplikasi penjadwalan produksi buku tulis pada jalur mesin 321 dengan menggunakan pendekatan *DBR* adalah bahwa jika terdapat perubahan kondisi sehingga perlu mengubah program aplikasi ini, sehingga dibutuhkan biaya yang cukup besar. Perubahan kondisi tersebut adalah jika perusahaan menambah jumlah mesin vurnish cover. Melihat hal tersebut, maka perlu pertimbangan dalam pengimplementasian program aplikasi penjadwalan ini di perusahaan.



BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil penelitian ini dapat diambil beberapa kesimpulan sebagai berikut:

1. Rancangan algoritma penjadwalan produksi buku tulis pada jalur mesin 321 terdiri atas penentuan *mean lead time* order dengan menggunakan algoritma Zijm-Buitenhek, penentuan *earliest bottleneck release* dan *bottleneck due date* order, pendistribusian order ke mesin di stasiun kerja *bottleneck*, pemeriksaan jadwal inisial mesin di stasiun kerja *bottleneck* (jika tidak layak maka dilakukan revisi jadwal inisial mesin dengan *Shortest Processing Time* dan jika layak maka dapat langsung dilakukan penjadwalan produksi). Penjadwalan produksi pertama kali dilakukan di stasiun kerja *bottleneck*, kemudian untuk stasiun kerja *upstream* dilakukan dengan *backward scheduling* dan untuk stasiun kerja *downstream* dilakukan dengan *forward scheduling*.
2. Rancangan program aplikasi penjadwalan produksi buku tulis pada jalur mesin 321 terdiri atas diagram alir program, *syntax* program dan *interface* program. Dengan penjadwalan produksi ini maka nilai *mean flowtime* order di lantai produksi bisa dikurangi hingga mencapai 62,86 % dengan menggunakan pendekatan *Drum-Buffer-Rope*.

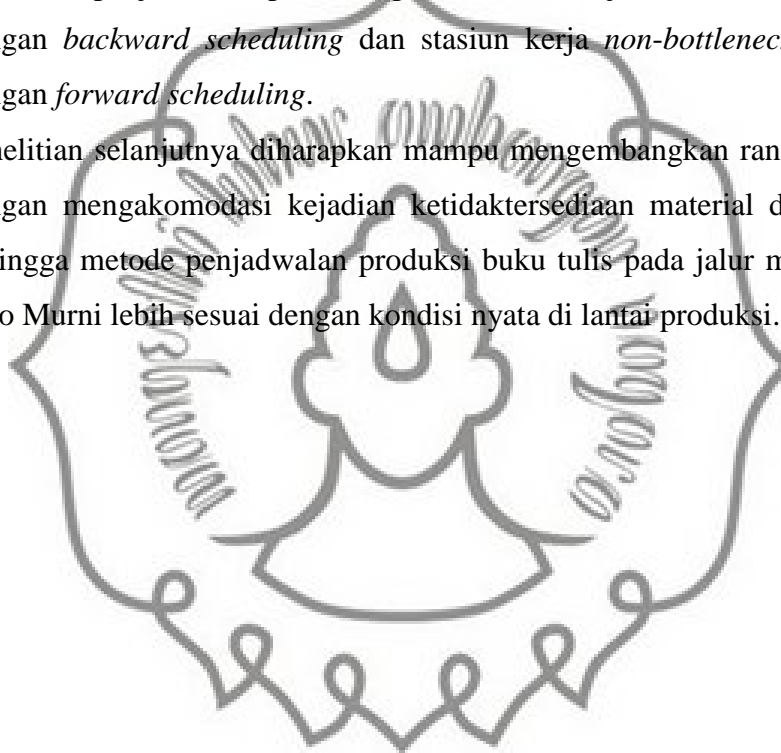
6.2 Saran

Berdasarkan hasil penelitian ini beberapa hal disarankan kepada PT. Solo Murni dan penelitian-penelitian selanjutnya. Saran-saran tersebut adalah:

1. Rata-rata *flowtime* order (*mean flowtime*) buku tulis jalur mesin 321 bisa dikurangi dengan penerapan rancangan algoritma penjadwalan produksi buku tulis yang melewati jalur mesin 321 dengan menggunakan pendekatan *Drum-Buffer-Rope*. Penjadwalan produksi buku tulis dengan menggunakan pendekatan *Drum-Buffer-Rope* perlu diperhatikan dalam penjadwalan

produksi buku tulis pada jalur mesin 321 sehingga daya kompetisi PT. Solo Murni di era persaingan bisnis saat ini bisa ditingkatkan.

2. Departemen PPIC yang melakukan penjadwalan harus memahami dan ada pelatihan terhadap program aplikasi penjadwalan produksi buku tulis yang diusulkan sehingga hasil yang diperoleh sesuai dengan yang diharapkan.
3. PPIC dalam menjadwalkan produksi buku tulis dengan rancangan usulan perlu memperhatikan penyesuaian yaitu penjadwalan produksi dilakukan dari stasiun kerja *bottleneck*, yaitu Stasiun Kerja Vurnish Cover, sebagai titik kontrol sistem. Berdasarkan jadwal produksi pada stasiun kerja *bottleneck* dilakukan penjadwalan produksi pada stasiun kerja *non-bottleneck upstream* dengan *backward scheduling* dan stasiun kerja *non-bottleneck downstream* dengan *forward scheduling*.
4. Penelitian selanjutnya diharapkan mampu mengembangkan rancangan usulan dengan mengakomodasi kejadian ketidaktersediaan material dan komponen sehingga metode penjadwalan produksi buku tulis pada jalur mesin 321, PT. Solo Murni lebih sesuai dengan kondisi nyata di lantai produksi.



DAFTAR PUSTAKA

- Baker, K.R. 1974. *Introduction To Sequencing and Scheduling*. New York: John Wiley and Sons.
- Bedworth, D.D. and Bailey, J.E. 1987. *Integrated Production Control Systems: Management, Analysis, Design*. Singapore: John Wiley and Sons.
- Dettmer, W.H. 1997. *Goldratt's Theory of Constraints "A Systems Approach to Continuous Improvement"*. Wisconsin: ASQC Quality Press.
- Fogarty, D.W., Blackstone, J.H and Hoffmann, T.R. 1991. *Production and Inventory Management*. Cincinnati: South-Western Publishing Co and APICS.
- Gazpersz, V. 2001. *Production Planning and Inventory Control*. Jakarta: Gramedia Pustaka Utama.
- Hillier, F.S. and Lieberman, G.J. 1995. *Introduction to Operations Research*. New York: McGraw-Hill.
- Iftadi, I. 2006. *Dasar-Dasar Algoritma dan Pemrograman Komputer*, Surakarta: UNS Press.
- Murthy, D.N.P., Page, N.W. and Rodin, E.Y. 1990. *Mathematical Modeling: A Tool for Problem Solving in Engineering, Physical, Biological, and Social Science*. Oxford: Pergamon Press.
- Narasimhan, S.L., McLeavey, D.W. and Billington, P.J. 1995. *Production Planning and Inventory Control*. New Jersey: Prentice Hall, Inc.
- Salvendy, G. 2001. *Handbook of Industrial Engineering "Technology and Operations Management"*. New York: John Wiley and Sons.
- Sipper, D. and Bulfin, R.L. 1997. *Production: Planning, Control, and Integration*. New York: McGraw-Hill.
- Tejaasih, I dan Suswamela. 2003. Aplikasi Model Penjadwalan *Drum Buffer Rope* Pada Sistem Manufaktur MTO Repetitif Dengan Sistem Produksi *Mixed Flow Shop*. *Prosiding Seminar Sistem Produksi VI* Hal 181- 196.
- Utomo, A.A. 2005. "Penjadwalan Produksi Minibus Jenis Varian Dengan Algoritma Penjadwalan *Drum-Buffer-Rope* Untuk Meminimasi *Mean Tardiness*". Skripsi. Jurusan Teknik Industri, Universitas Sebelas Maret Surakarta.

www.wikipedia.com/wiki_Visual_Fox_Pro

Zijm, W. H. M., and Buitenhek, R. 1996. "Capacity Planning and Leadtime Management". *International Journal of Production Economics*, 46. pp 165-179.

